

Mihail Radu Solcan

Tehnici de
elaborare
a manuscrisului
și elemente
de editare
și tehnoredactare
computerizată

Eseul filosofic



Editura
Universității din București

Mihail Radu Solcan

Eseul filosofic

Mihail Radu Solcan

Tehnici de elaborare
de editare și tehnoredactare
computerizată
a manuscrisului și
elemente de editare și tehnoredactare
computerizată
Eseul filosofic



Editura
Universității din București
2004

Referenți științifici:

prof. univ. dr. Adrian-Paul ILIESCU

lect. univ. dr. Romulus BRÂNCOVEANU

© Editura Universității din București

Șos. Panduri, 90-92, București – 050663; Telefon/Fax: 410.23.84

E-mail: editura@unibuc.ro

Internet: www.editura.unibuc.ro

Descrierea CIP a Bibliotecii Naționale a României
SOLCAN, RADU M.

**Eseul filosofic: Tehnici de elaborare a manuscrisului
și elemente de editare și tehnoredactare
computerizată / Mihail Radu Solcan – București:**
Editura Universității din București, 2004

Bibliogr.

ISBN 973-575-880-6

665.254.2

Avertisment

Textul acestei cărți reflectă experiența autorului. El reflectă, de asemenea, o serie de preferințe ale autorului. Orice recomandare făcută aici trebuie privită ca o recomandare adresată studentelor sau studenților care elaborează lucrări de seminar, lucrări de diplomă sau teze de masterat sub conducerea sa.

Dacă urmați regulile formulate aici la alte cursuri sau seminarii decât cele ale autorului, o faceți pe răspunderea dumneavoastră. Cred c-ar fi cel mai bine să consultați cadrele didactice îndrumătoare atunci când aveți dubii asupra unei soluții sau a alteia, asupra unui anumit standard sau oricărui alt aspect al cărții de față.

Este la fel de important să înțelegeți că toate explicațiile referitoare la folosirea computerului sunt rodul practicii personale a autorului și oglindesc în mod inevitabil și limitele acestei practici. Toate soluțiile și programele descrise în carte au fost testate de către autor, inclusiv pe parcursul elaborării cărții ca atare. Autorul și editura nu-și asumă însă nici un fel de răspundere directă sau indirectă pentru eventuale neajunsuri sau daune care ar putea rezulta din folosirea cărții sau a programelor de calculator prezentate în carte.

Utilizarea programelor și a soluțiilor descrise în carte trebuie făcută de cititoare sau cititori pe propriul lor risc.

Sprijin în utilizarea programelor sau aplicarea soluțiilor prezentate în carte nu pot primi, în limita timpului disponibil, decât studențele sau studenții care elaborează lucrări sub îndrumarea autorului. Orice alte persoane trebuie să ia textul cărții ca atare, fără posibilitatea de a cere sprijin sau îndrumări în aplicarea soluțiilor sau utilizarea programelor.

În sfârșit, autorul nu se angajează sub nici o formă să asigure sprijin în dezvoltarea de noi programe sau soluții de aplicare a unor programe pentru nici o categorie de persoane.

Multe denumiri sunt folosite de către firme sau creatorii de pro-

grame individuali pentru a distinge produsele lor. Unele dintre ele sunt menționate în carte, respectând forma proprie denumirii respective. Cititoarele și cititorii trebuie să știe că aceste denumiri nu pot fi utilizate de către alte firme sau persoane individuale decât pentru desemnarea produselor respective.

Pentru alte precizări și informații suplimentare consultați situl pe Internet al cărții la <<http://www.fil.unibuc.ro/~solcan/eft>>.

Cuprins

Avertisment	v
Mulțumiri	ix
I Furtuna de idei	1
1 Tehnologia muncii intelectuale	3
1.1 Sursele eseurilor	4
1.2 Fișele	12
1.3 În căutarea ideilor-cheie	20
1.4 Revoluția computerelor și munca intelectuală	24
1.5 Computerul pregătit pentru munca intelectuală	29
2 La început a fost proiectul	31
2.1 Proiectul de cercetare	32
2.2 Documentarea	36
2.3 Publicul cititor	39
2.4 Construiți proiectul cu ajutorul computerului	41
II Șantierul	43
3 Modularitatea manuscrisului	45
3.1 Secretul este rescrierea lucrării	46
3.2 Asamblarea modulelor	48
3.3 Modularizarea computerizată	50
4 Conținutul textului	51
4.1 Eseul ca încercare de a dezlega o problemă	52
4.2 Limbajul filosofic	55
4.3 Argumentarea filosofică	61
4.4 Dezvoltarea conținutului	65
4.5 Ajutorul limitat al computerului	66
5 Forma textului	67
5.1 Bibliografia	67
5.2 Notele	70
5.3 Trimiterile interne	76
5.4 Indexarea eselui	77
5.5 Lăsați \LaTeX să aibă grijă de formă	79

III	Arta supraviețuirii	81
6	Cu ochii redacției	83
6.1	Stilul eseului filosofic	83
6.2	Corectura	108
6.3	Computerul și faza finală a elaborării textului	112
7	Apărarea și evaluarea eseului	113
7.1	Susținerea eseului	113
7.2	Evaluarea eseului	114
7.3	Eseuri trimise altor universități	117
7.4	Computerul și susținerea eseului	118
	Anexe	121
A	Creionul electronic	121
A.1	Sistemul de operare și managementul fișierelor	121
A.2	Editorul Vim	144
A.3	Expresiile regulate	166
A.4	În căutarea surselor	172
A.5	Corectura computerizată	180
B	Tehnoredactarea computerizată	183
B.1	\LaTeX	183
B.2	\BIBTeX	233
B.3	Turnul Babel	240
B.4	Tabele și formule	263
B.5	Indexarea electronică	275
C	Pensula electronică	277
C.1	Imaginile	277
C.2	Inserarea imaginilor în \LaTeX	282
C.3	Inserarea literelor ca inserare de imagini	284
D	Translatorii	289
D.1	Drumul către HTML	290
D.2	Drumul către RTF	296
D.3	Înapoi către \LaTeX	297
	Bibliografie	299
	Index	303

Mulțumiri

Sunt foarte multe persoane cărora le sunt îndatorat. Recunoștința mea se îndreaptă în primul rând către profesorul Mircea Flonta, îndrumătorul tezei mele de doctorat. De la domnia sa am deprins gustul pentru stilul filosofic sobru și clar, distinct de cel literar, ideea de a separa net textul de bază de note, atenția pentru liniile de forță ale textului, distincția dintre problemele pe care le-ai tratat și cele rămase deschise și multe altele.

Tatăl meu, Pan Solcan, scriitor și ziarist, m-a dus prima oară într-o redacție și o tipografie. Am avut astfel norocul să ajung acum peste patruzeci de ani în contact direct cu lumea celor care scriu, redactează, tehnoredactează, culeg, corectează și tipăresc texte. Tatăl meu m-a inițiat și-n tainele scrisului literar. Din păcate, n-am cultivat în mod sistematic arta scrisului literar.

În anii '70 și '80 am lucrat ca documentarist. Am învățat o mulțime de lucruri de la colegele și colegii mei de atunci. Fără doamna Eva Țuțui aș ști mult mai puține despre traducere, trimiteri și multe alte lucruri ce țin de redactarea textelor academice. Nicolae Frigioiu te contamina cu pasiunea lui pentru cunoașterea din cele mai diferite domenii. Nicolae Lotreanu a fost nu doar un șef îngăduitor, ci și omul de la care am învățat cât este de important să scrii un rezumat neutru, fără comentarii și judecăți de valoare.

Am revenit în contact cu lumea editurilor la începutul anilor '90, în condițiile extraordinare ale descătușării de constrângerile regimului totalitar. Am o mulțime de datorii față de colegii de la editura Humanitas. Caietul cu *Normele de editare Humanitas* este și acum un standard pe care-l consult cu atenție. Domnul Gabriel Liiceanu a adus în editură pasiunea sa pentru textul redactat perfect. Domniei sale îi datorez extraordinara posibilitate de a mă documenta la fața locului în lumea editurilor universitare americane.

În Facultatea de filosofie, profesorii Adrian Miroiu, Valentin Mușan și Mircea Dumitru au fost decanii care au pus bazele și au consolidat un sistem de lucru și de examinare în care eseul filoso-

fic joacă un rol foarte important. Din punctul de vedere al formei procesului de învățământ, accentul pe eseu cred că este ceea ce distinge în mod pregnant Facultatea de filosofie a anilor '90 de cea din perioada anterioară.

Colegii din Catedra de filosofie politică și morală m-au ajutat cu sugestiile și observațiile lor. Adrian Iliescu și Romulus Brâncoveanu m-au sprijinit în conturarea precisă a conținutului și limitelor ariei de aplicare a regulilor din cartea de față. Valentin Mureșan insistă de multă vreme asupra necesității de a stabili standarde precise pentru forma eseurilor. Întrebările sale referitoare la natura plagiatului, modul de a introduce citatele și multe altele mi-au fost de un mare folos.

Din păcate, nu este suficient spațiu pentru a menționa persoanele care m-au ajutat decisiv în structurarea ideilor din cartea de față: studentele și studenții mei și colegii mai tineri care au condus seminariile de la cursurile mele. Îmi cer scuze pentru faptul că-i menționez în mod colectiv. Este vorba însă despre sute și sute de eseuri, de sute de cazuri discutate.

Existența acum a unui text scris sper să fie de ajutor în conturarea precisă a regulilor redactării și tehnoredactării eseului.

Sunt recunoscător tuturor persoanelor pe care le-am menționat direct sau indirect, dar trebuie spus că ele nu poartă absolut nici o vină pentru neajunsurile cărții. Nimeni nu trebuie, de asemenea, să considere că persoanele menționate aici ar avea vederi care coincid integral cu ideile exprimate în carte. De la caz la caz, sunt posibile dezacorduri, uneori substanțiale. Înainte de a utiliza cartea, consultați *Avertismentul* plasat în deschidere.

Așa cum o cer regulile jocului academic, ajutorul primit de-a lungul timpului trebuie recunoscut, dar trebuie să-ți asumi răspunderea pentru ceea ce-ai scris. Pentru limitele între care este plasată această răspundere, din pricina specificului cărții de față, trebuie din nou consultat *Avertismentul* plasat la începutul cărții.

Partea I

Furtuna de idei

Capitolul 1

Tehnologia muncii intelectuale: tradiție și modernitate

Cuprins

1.1	Sursele eseurilor	4
1.1.1	Sursele primare	5
1.1.2	Sursele secundare	7
1.1.3	Clasificarea surselor	7
1.2	Fișele	12
1.2.1	Formatele fișelor	12
1.2.2	Fișele bibliografice	13
1.2.3	Fișele de lectură	17
1.2.4	Fișele de lucru	20
1.3	În căutarea ideilor-cheie	20
1.3.1	Reconstituirea argumentelor	23
1.4	Revoluția computerelor și munca intelectuală	24
1.4.1	Dilema muncii intelectuale computerizate	25
1.4.2	Investiții într-un computer	27
1.5	Computerul pregătit pentru munca intelectuală	29

Termenul de „eseu“ poate fi foarte derutant. Următoarea întâmplare ilustrează limpede ambiguitatea acestui termen. Un student a venit la unul dintre examenele mele și mi-a explicat cum și-a conceput lucrarea: a luat o carte, a citit-o; i-au venit idei; a dat cartea deoparte și a scris. Lucrarea sa nu avea bibliografie, note de subsol și trimiteri. Era un exercițiu de imaginație și reflecție.

Ceea ce înțelegem noi aici prin eseu este exact opusul sensului avut în vedere de studentul din povestea de mai sus. Eseul trebuie să aibă bibliografie, note de subsol, trimiteri, secțiuni numerotate

și, dacă dimensiunile sale sunt mai ample, indici. Analiza din eseu este o încercare de a înțelege o discuție din literatura de specialitate. Ipotezele personale reprezintă o intervenție în discuția respectivă și trebuie susținute cu argumente.

De ce am folosit termenul de „eseu”? În ultimul deceniu, el a fost utilizat curent în Facultatea de filosofie a Universității din București pentru a desemna lucrările de seminar și lucrările prezentate la examene. Am extins aici sfera de utilizare a acestui termen, dar am adăugat determinativul „filosofic” pentru mai multă claritate.

Tehnicile descrise în carte sunt aplicabile în orice eseu academic. Nu există însă suficiente detalii pentru ca studenta sau studentul de la matematică să scrie formule complicate, studenta de la lingvistică să deseneze diagrame, cea de la istorie sau de la psihologie să introducă tabele statistice sofisticate sau persoanele interesate de chimie să producă formulele specifice respectivei discipline. Toate aceste persoane ar putea folosi tehnicile descrise în carte pentru a trasa cadrul de bază al lucrărilor de specialitate. Persoanele interesate de filosofie vor găsi însă suficient material pentru a scrie orice tip de eseu filosofic.

Dacă încercăm să scriem un eseu în sensul precizat mai sus ne trezim într-o lume extrem de agitată. Lumea ideilor este departe de a fi o lume în care este ușor să te orientezi.

Termenul englezesc de *brainstorming*¹ se referă la o metodă de organizare a dezbatelor, în care intervenția neîngrădită a participanților ar trebui să conducă la idei noi.

dezbate-
rea ca
semn al
existenței
unei
probleme

Noi am folosit termenul de „furtună a ideilor”, inspirat din cel de *brainstorming*, pentru a sugera starea agitație pe care o veți găsi în literatură atunci când se dezbate o problemă. Intenția noastră este însă nu aceea de a trimite la o metodă de a produce idei, ci de a atrage atenția asupra necesității unor hărți și instrumente intelectuale care să ne ajute să nu naufragiem.

1.1 Sursele eseurilor

Unele persoane cred că orice sursă de idei, experiența lor de viață, trăirile lor în contactul cu o operă de artă și altele, este un bun punct de plecare pentru un eseu filosofic. La mijloc este o mare confuzie, pe care am descris-o deja în deschiderea acestui capitol.

Punctul de plecare îl constituie o dezbatere din literatura de spe-

¹ *Ad litteram* înseamnă „furtuna creierelor”.

cialitate. Principiul acesta este valabil pentru orice eseu academic. Dificultatea rezidă însă în diversitatea amănunțită de opinii, îndeosebi în lumea filosofiei.

Pentru unii autori, care merg pe urmele lui Karl Popper, gândirea lui Platon este cea care a înstructurat concepțiile de tip totalitar. Pentru alți autori nici nu poate fi vorba despre o astfel de interpretare a lui Platon.

Mai mult decât atât, fiecare autoare sau autor speră, în adâncul sufletului, să contribuie la furtuna ideilor din lumea filosofiei. Prețul diversității este însă chiar marea varietate de informații. Avem nevoie de unelte pentru a manevra aceste informații extrem de variate și pentru a ne orienta într-un peisaj derutant de complicat.

dezbaterea ca punct de plecare al eseului

1.1.1 Sursele primare

Scopul principal al acestei cărți este să ofere soluții practice. De aceea nu căutați aici o discuție teoretică despre distincția dintre sursele primare și cele secundare.² Principiul acesta se aplică, de altfel, și-n toate secțiunile care urmează.

Sursele primare pentru eseurile filosofice sunt textele în care autorii sau autoarele își susțin propriile opinii.

Dacă un autor relatează ceea ce a spus altul. Dacă ne dă chiar citate din scrierile altora trebuie să tratăm respectivul document ca pe unul secundar.

Același text ar putea fi însă citit în două feluri. Nu ne putem face o idee despre ce a zis Platon din rezumatele altora. Putem însă să ne facem o idee despre modul în care l-au citit ei pe Platon.

Ce ne facem însă atunci când, mai ales în cazul autorilor vechi, nu mai avem la dispoziție decât relatările altora? În această situație este nevoie de exercitarea spiritului nostru critic. Dacă este posibil, trebuie să comparăm mai multe surse. Dacă n-avem decât o singură sursă, trebuie să ne facem o idee cât mai exactă despre „grila de lectură” folosită în sursa de care dispunem.

²Mihailov, Ceornii și Ghilearevski[28, pp.85–181] tratează pe larg problemele documentelor primare și secundare, așa cum se puneau ele la mijlocul secolului trecut. Personal am îndoieli cu privire la valoarea strict teoretică a unor astfel de discuții. Partea practică este însă importantă.

1.1.1.1 Citiți textul în original

studiați originalul

În ce limbă ar trebui să citim documentele primare? Răspunsul la această întrebare nu poate fi decât unul, dacă doriți să scrieți o lucrare de specialitate. Textele filosofice trebuie citite în limba în care au fost scrise.

La ce sunt bune traducerile? Aici se aplică un principiu valabil și-n cazul dicționarilor. Dacă studiați, de pildă, un text în limba engleză, consultați mai întâi un dicționar explicativ al limbii engleze. După ce vă faceți o idee despre ce este vorba, consultați un dicționar englez-român. Recurgeți la dicționarul bilingv doar pentru a găsi o soluție mai bună în limba română. Tot așa, în cazul traducerilor, recurgeți la ele abia după ce v-ați făcut o idee despre original.

Ce facem dacă nu știm limba în care a fost scris textul? În acest caz, cel mai bun sfat este să consultați cât mai multe traduceri. Comparați aceste traduceri și veți avea o idee destul de exactă despre textul de bază. Multă lume procedează așa, de pildă, în cazul limbii grecești vechi.

Ce limbi moderne sunt importante pentru filosofie? Indubitabil, trei limbi de mare circulație joacă un rol deosebit: engleza, germana și franceza. Italiană este însă și ea deosebit de utilă pentru cei care traduc în limba română. Soluțiile traducătorilor italieni cred că sunt deosebit de importante pentru cine traduce în limba română. De asemenea, modul lor de a traduce este mult mai precis decât cel din Franța. În Franța, tradiția traducerii cere să jertfești precizia de dragul stilului frumos.

Înainte de 1990, cei interesați de logică erau foarte câștigați dacă aveau acces la limba rusă. Aproape toate scrierile importante din literatura de specialitate sunt traduse în limba rusă. O altă limbă slavă, poloneza, poate fi, de asemenea, indispensabilă pentru persoanele care vor să studieze aprofundat marea școală de filosofie și logică din Polonia.

1.1.1.2 Antologiile

Am auzit de multe ori părerea că textele filosofice nu pot fi decupate în vederea includerii într-o antologie. Opinia mea este total opusă. Textele filosofice nu sunt romane. Este perfect rezonabil să decupăm o secțiune care prezintă interes pentru tema antologiei.

Antologiile care reproduc textele în limba în care au fost scrise sunt surse excelente. Le putem trata ca pe niște surse primare. În fond, dacă simțim nevoia, vom consulta oricum textul integral.

Dacă o antologie este bine făcută, ea oferă deja teme, o introducere, bibliografii și, evident, textele de bază. Dacă textele antologate sunt redactate în traducere, cred că – din punctul de vedere al persoanei care se specializează în domeniul respectiv – trebuie tratate drept surse secundare.

1.1.2 Sursele secundare

Sursele secundare, vorbind strict din perspectiva informaticii documentare sunt enciclopediile, bibliografiile, diversele ghiduri.

După cum arătam în secțiunea 1.1.1.1, este bine să tratăm și orice traducere ca pe o sursă secundară. Evident, acest mod de a privi lucrurile este valabil doar din perspectiva persoanei care vrea să scrie un eseu de specialitate. Pentru persoanele care vor doar să guste din plăcerea de a citi un text traducerile sunt poate chiar mai potrivite decât originalul.

Există o sumedenie de ghiduri pentru filosofie.³ De la caz la caz, în funcție de tema eseului, trebuie văzut care este cel mai util.⁴

Ghidurile pe care le consultăm când scriem un eseu sunt ca orice ghid de călătorie. Nu este bine să plecăm la drum fără un ghid de călătorie. Ar fi însă fără sens ca, odată ajunși la destinație, să rămânem cu ochii ațintiți la paginile ghidului și să nu admirăm pe cont propriu peisajul sau să nu ne aventurăm pe drumurile mai deosebite despre care ne relatează ghidul.

**ghiduri
pentru
elabora-
rea
lucrărilor
studen-
tești**

1.1.3 Clasificarea surselor

Aveți enorm de câștigat dacă înțelegeți sistemul de clasificare al bibliotecii. Nu se pune problema de a deveni specialistă sau specialist în biblioteconomie. Înțelegerea principiilor pe care se bazează clasificarea, precum și a liniilor ei de forță ne oferă însă o hartă a cunoștințelor omenești.

Ni se pare normal să avem cunoștințe de geografie. Facem chiar glume pe seama celor care încurcă între ele capitale din Europa.

³Câteva dintre aceste îndrumare, disponibile pe Internet sunt incluse în bibliografia finală a acestei cărți. Consultați în special pe cele scrise de către Martin Young[47], James Pryor[36], Douglas Portmore[35], Peter Suber[42], Peter Horban[18] sau Ashley McDowell[25].

⁴Dacă scrieți, de pildă, un eseu la etica aplicată, ghidul lui Barry Brown[6] pentru lucrări pe teme din bioetică poate fi mai folositor decât un îndrumar cu caracter general.

Din păcate, în școală, nu se predă absolut nimic despre hărțile cunoștințelor omenesti.

Lacuna semnalată mai sus nu poate fi eliminată prin informațiile sumare oferite în continuare. Le puteți însă folosi ca pe niște capete de fir. Dacă trageți de ele, cu puțin efort, veți afla mult mai multe despre geografia cunoașterii umane.

Atenție! Nu confundați sistemul de clasificare, care este o hartă bazată pe conținutul publicațiilor cu cotele folosite în biblioteci pentru a identifica poziția cărții pe rafturile bibliotecii. Cotele formează o hartă bazată pe poziția cărții în rafturi și, uneori, pe caracteristicile ce țin de tehnoredactare (dimensiunile cărții, de exemplu).

Cărțile dintr-o bibliotecă mai au pe ele și un număr de inventar. Acest număr este important pentru gestiunea fondului de carte.⁵

De ce am clasifica sursele bibliografice? Închipuiți-vă că aveți în grijă o bibliotecă în care se adună, de-a lungul timpului, mii de volume, poate zeci de mii de volume. Le aveți în gestiune și le-ați dat un număr de inventar. Le atașați o cotă care vă permite să identificați locul din bibliotecă în care se află cartea. Ar fi greu de ținut minte unde se află mii de volume. De asemenea, fără aceste cote ar fi greu de pus la loc volumele.

Este însă imposibil să specializați anumite rafturi pentru o temă. Noile achiziții nu vor încăpea, de la un timp, în zona temei respective. Și mai grav este însă faptul că un volum ar putea figura la mai multe teme. *Republica* lui Platon ar putea fi pusă în zona rezervată filosofiei politice, dar și-n cea a filosofiei minții etc. O altă carte ar putea prezenta un interes matematic intrinsec, dar ar putea fi o sursă și pentru filosofia matematicii.

**rolul
clasificării
cărților**

Soluția este un sistem de clasificare care nu implică așezarea fizică a cărților într-o anumită zonă. Fiecare carte primește un cod, un fel de cheie, cu ajutorul căruia putem afla temele pentru care este relevantă cartea. Fișele bibliografice ale cărților au pe ele aceste coduri, dar au și cotele care permit localizarea pe raft a cărții.

Construcția unui sistem de clasificare nu este nici pe departe un lucru simplu. Vom descrie doar ideile de bază ale sistemelor de clasificare mai răspândite.

⁵Dacă găsiți abandonată pe undeva o carte cu număr de inventar și sigla bibliotecii, duceți cartea înapoi. Cineva o are în gestiune și răspunde financiar pentru ea. Atrageți atenția oricui ar vrea să distrugă sau să sustragă o carte din bibliotecă asupra răului pe care-l face persoanei care are cartea în gestiune. Cred că acest rău este ceva mult mai inteligibil decât cel la care ar face apel retorica vagă a răului făcut celor însetați de cunoaștere.

1.1.3.1 Dewey

Americanul Melvin Dewey a publicat, în 1876, o schemă de clasificare care avea să exercite o influență enormă. Ideea de bază a lui Dewey este simplă. Toate cărțile formează o mare mulțime, sunt trunchiul unui arbore uriaș. Mai întâi acest arbore este desfăcut în zece ramuri. Apoi fiecare ramură este desfăcută în alte zece ramuri și procesul poate continua.

Am consultat versiunea prescurtată a clasificării lui Dewey[8], o versiune potrivită pentru bibliotecile mici.⁶ Am și folosit-o pentru biblioteca personală.

Ramurile sunt de către Dewey cu cifre. De exemplu, 111 nu înseamnă altceva decât că parcurgem structura arborescentă în felul următor: filosofie → metafizică → ontologie.

Logica are codul 160. Cifra 0 este semnul pentru ramura clasificării unde sunt puse lucrările cu caracter general. Dacă vrem să dăm de o ramură anume a logicii, atunci trebuie să mergem mai departe. Codul pentru logica simbolică este 164.

Cifrele codurilor sunt grupate câte trei și se utilizează un sistem judicios de analogii pentru a extinde clasificarea. De exemplu, codul 199 este pentru „alte filosofii moderne“. Aici ar intra și filosofia românească. Pentru a-i stabili exact codul ne uităm la modul în care este divizată ramura 9, cea a istoriei. Obținem pentru filosofia românească modernă codul 199.498.

1.1.3.2 Biblioteca Congresului SUA

Biblioteca Congresului SUA este una dintre cele mai mari biblioteci din lume. Ea are propriul ei sistem de clasificare.

În sistemul Bibliotecii Congresului trunchiul cel mare al cunoașterii umane este despiciat în ramuri care au drept cod o literă. Codul filosofiei este B.

Ramurile care se desfac din B primesc tot o literă. Logica are, de pildă, codul BC. În limbajul tehnic al clasificării respective, aceasta este o *subclasă*.

Fiecare subclasă codificată prin litere constituie un spațiu care este divizat mai departe și codificat cu cifre. De exemplu, între BF309 și BF499 se află tot ce este legat de conștiință și cogniție, incluzând

⁶Consultați pagina de web <<http://www.oclc.org/fp/>> a editurii Forrest Press pentru a afla mai multe despre clasificarea Dewey.

aici orice text pe teme precum inteligența, gândirea, memoria, psiholingvistica.

Biblioteca Congresului este accesibilă pe Internet⁷ și găsiți pe situl ei o introducere în sistemul ei de clasificare.⁸ Puteți să testați cunoștințele dobândite răsfoind prin intermediul Internetului imensul catalog al Bibliotecii Congresului.

1.1.3.3 Clasificarea zecimală

În bibliotecile noastre este folosită Clasificarea Zecimală Universală (CZU). Principiul ei de bază este același ca și cel din clasificarea Dewey. Practic, CZU este forma internațională, tradusă într-o mulțime de limbi a clasificării Dewey.

CZU folosește semne auxiliare pentru a marca relațiile între teme. De pildă, două puncte denotă o relație. Semnul egal precede indicele pentru limbă.

CZU a fost utilizată pe vremea când se sortau cartelele cu mașini mecanice; mai nou, a fost adaptată la folosirea în programe de calculator. Adaptarea realizată de către UDC Consortium nu îndeplinește însă criteriile stipulate în anexele acestei cărți pentru programele utilizate în mediul academic și de aceea nu este prezentată acolo.⁹

Nu este nici o problemă însă să adăugați un câmp czu într-un fișier de tip bib și puteți folosi CZU cu BIBTEX.¹⁰

1.1.3.4 Descrierea cărților de către editori

Dacă răsfoiți cărți publicate cu două-trei decenii în urmă la noi, nu veți găsi decât o casetă tehnică la sfârșit și cam atât. Nici un fel de numere speciale pentru carte. Am căutat destul de mult prin bibliotecă pentru ca să dau peste o carte cu indicații de clasificare. Istoria logicii scrisă de William și Martha Kneale¹¹ are în caseta tehnică un cod CZU. Codul respectiv este 16. Chiar și pe baza explicațiilor noastre sumare de mai sus, este ușor de înțeles ce spune acest cod. Ne spune că este vorba despre o carte de logică și atât.

După 1989, editurile noastre se străduiesc să intre în circuitul

⁷V. adresa <<http://www.loc.gov>>.

⁸V. pagina de web <<http://www.loc.gov/catdir/cpsol/lcco/lcco.html>>.

⁹Consultați pagina de web <<http://www.udcc.org/mrf.htm>> pentru a descrierea soluției UDC Consortium și licența de utilizare.

¹⁰Consultați §B.2 pentru sistemul BIBTEX.

¹¹William și Martha Kneale, *Dezvoltarea logicii* (Cluj-Napoca: Dacia, 1974).

internațional. Nu s-ar putea spune că s-au îmbunătățit peste tot descrierile cărților, dar cred că nu mai există editură care să nu includă „ISBN“.

1.1.3.4.1 ISBN ISBN (International Standard Book Number) este un sistem internațional de numerotare a cărților. Ideea a fost lansată în Marea Britanie de către marea rețea de distribuire de cărți W.H.Smith și de către asociația editorilor, în 1965.

În România există un centru național ISBN.¹² La acest centru puteți găsi și o listă a codurilor ISBN.

Dacă știți codul ISBN al unei cărți puteți căuta cu mare ușurință pe Internet date despre cartea respectivă. Editurile și marile librării folosesc acest cod.

1.1.3.4.2 ISSN ISSN (International Standard Serial Number) este pentru publicațiile periodice ceea ce este ISBN pentru cărți. Într-o lume în care numărul publicațiilor tinde să devină uriaș, dacă știți ISSN-ul unei publicații periodice, aveți mari șanse să găsiți informații despre ea cu ajutorul Internetului.

Există un centru român pentru ISSN și acest centru are o pagină de web.¹³

1.1.3.4.3 CIP CIP (Cataloguing in Publication) este un sistem voluntar de cooperare între editori și biblioteci. El permite ca descrierea cărții să fie inclusă deja inclusă în carte. Editura Universității din București folosește acest sistem.

CIP-ul include numele autoarei sau autorului, ISBN-ul, cuvinte-cheie pentru temele cărții, informații cu privire la existența bibliografiei, a indicilor.

De la țară la țară, elementele CIP-ul pot să difere. De regulă, CIP-ul include și coduri pentru CZU, Dewey sau clasificarea Bibliotecii Congresului sau coduri specifice țării respective. În publicațiile românești, se folosește CZU.

¹²Informațiile prezentate aici sunt extrase din pagina de web a acestui centru <<http://www.bibnat.ro/produse/ISBN.htm>>, vizitată pe data de 20/08/2003.

¹³V. <<http://www.bibnat.ro/produse/ISSN.htm>>.

1.2 Instrumentul de bază al muncii intelectuale: fișele

Cuvântul-cheie pentru tehnicile tradiționale de muncă intelectuală este „fișă”. Dulapurile cu fișe erau un element nelipsit din biblioteci înainte de folosirea masivă a calculatoarelor.

Când eram tineri, soția mea și cu mine am cumpărat un dulap numai cu sertare. Intenția era de a-l folosi pentru depozitarea de fișe. Ocupă și acum o jumătate de hol, dar este folosit în mică măsură pentru destinația inițială.

**rolul
fișelor tra-
diționale** Cred că nu este bine ca fișele tradiționale să fie total eliminate de dischete și CD-uri. În primul rând, dacă știi să lucrezi cu ele, atunci înțelegi mult mai ușor ce se întâmplă cu adevărat când prelucrezi informații. În al doilea rând, poți realiza fișe chiar și atunci când n-ai calculatorul cu tine, le poți întinde prin toată casa și le poți afișa pe panouri speciale. Chiar și aceste rânduri sunt scrise pe baza ideilor schițate pe o fișă de lucru tradițională.

1.2.1 Formatele fișelor

Fișele au diverse formate, în funcție de destinația lor. Ar fi inutil să indicăm dimensiunea fișelor în milimetri. În mod sigur, hârtia pe care o folosiți va avea, de la un producător la altul, dimensiuni ușor diferite. Vom lua drept reper coala de hârtie denumită A4.¹⁴

Formatul A4 este cel folosit pentru lucrări de seminar, lucrări de diplomă, disertații și teze de doctorat.

Formatul A3 este dublul lui A4. Copiatoarele mai mari pot, de pildă, lucra cu un asemenea format. A3 este potrivit pentru afișe. În munca intelectuală ar putea fi folosit atunci când vreți să desenați o rețea cu relații complicate între capitole și secțiuni sau diagrama unui argument foarte complicat.

În nici un caz A3 sau A4 nu sunt formate pentru fișele uzuale. Proiectul de cercetare este normal să fie lucrat pe un format A4 pentru că el este, în fond, ca un studiu preliminar sau un raport de cercetare. Are alte funcții decât lucrarea finală, dar nu există motive speciale pentru a-i da un format aparte.

Fișele, în schimb, vrem să le aranjăm, să le permutăm. Unele dintre ele am vrea să stea drepte în cutii sau în sertare. Fizic, nu

¹⁴Manualele mai vechi dedicate tehnicilor muncii științifice puneau un mare accent pe fișe. Formatele fișelor sunt descrise de Heyde[17, pp.18-23.].

prea are sens să folosim hârtii de format prea mare, care se vor îndoi ușor în sertar sau în cutia în care le ținem.

Formatul A5 reprezintă jumătatea lui A4. Luați o foaie de hârtie normală, împărțiți în două lungimea ei, tăiați și obțineți o fișă format A5.

Pentru a obține o fișă de format A6, divizați în două o fișă de format A5, după metoda descrisă mai sus. Cu aproximație, formatul obținut este cel al unei cărți poștale obișnuite.¹⁵ Formatul acesta va juca un rol foarte important în cele ce urmează.

**formatul
tip carte
poștală**

După aceeași metodă, din A6 obțineți o fișă format A7. Tot cu aproximație, am putea asemui această fișă cu fișele folosite în mod tradițional în biblioteci pentru a descrie cărțile (descrierea bibliografică).

Fișa cea mai mică este cea de format A8. Poate fi folosită pentru a nota un cuvânt sau o expresie scurtă și câteva numere, ca atunci când construim indexul unei cărți.

1.2.2 Fișele bibliografice

Fișele bibliografice au un rol foarte simplu, dar extrem de important. Cu ajutorul lor alcătuim ulterior bibliografia. De asemenea, sunt necesare pentru a nu consemna în mod repetat toate datele în legătură cu un anumit document.

În mod normal, formatul unei fișe bibliografice este A7. Cred că nu este însă o greșală dacă folosim o fișă tip „carte poștală” și pentru descrierea bibliografică a unui document.

Pentru a înțelege exact de ce date avem nevoie când descriem un document este bine să avem permanent în minte un principiu fundamental. Descrierea noastră va fi folosită până la urmă pentru a indica cititoarei sau cititorului drumul pe care să meargă pentru a verifica o anumită informație. Calea către informația relevantă nu trebuie indicată în mod ambiguu. Dacă aș spune „în filosofia analitică”, această trimitere este absolut neclară și nerecomandabilă într-o scriere academică.

**principiul
regăsirii
informa-
ției**

1.2.2.1 Fișele bibliografice pentru cărți

Descrierea unei cărți de autor netraduse este exemplul cel mai simplu de descriere a unui document pe care-l putem găsi într-o bibliotecă.

¹⁵Heyde[17, p.23].

Cartea are un autor sau o autoare. Are un titlu. Ea a fost publicată într-un anumit mod. Simplificând, putem spune că ne interesează trei mari blocuri de date:

Date despre autoare/autor/editor/editoare Consemnați numele persoanei sau persoanelor care au scris cartea. Mai rar, s-ar putea ca o organizație să-și asume calitatea de autoare. Dacă este vorba despre o culegere de studii, trebuie identificat un editor sau o editoare.

Date despre text Cartea are un titlu. Dacă este vorba despre o culegere, trebuie făcută o distincție clară între titlurile studiilor și cel al întregii culegeri. În fișa volumului notați *doar* titlul volumului și atât.

Date privind publicarea cărții Localitatea, editura și anul publicării.

Nu este rău să consemnați o serie de date suplimentare. Cartea are un cod în diversele clasificări. Poate avea un ISBN. Dacă ați găsit-o într-o bibliotecă publică, atunci are o cotă. O descriere CIP v-ar ajuta să descoperiți cuvinte-cheie pentru cartea respectivă. Puteți folosi un câmp cum este câmpul pentru cuvinte-cheie din figura 1.1 pentru a reține aceste informații.

Dacă folosiți fișa din figura 1.1 ca model, adaptați câmpul principal (cel care acolo servește la consemnarea propoziției-cheie) la nevoile legate de cele trei blocuri fundamentale de informații bibliografice.

cazul traducerilor Nu cred că trebuie încărcate cele trei blocuri de informații cu prea multe informații suplimentare față de cele care sunt strict necesare pentru regăsirea *sursei*. Dacă este vorba de o traducere, informațiile cu privire la text trebuie însă completate prin includerea numelui traducătoarei/traducătorului. De asemenea, unele ediții sunt îngrijite de către un specialist. Îngrijirea ediției trebuie menționată mai ales când este vorba efectiv despre o ediție critică (cu note, consemnarea variantelor din manuscrise și așa mai departe).

În cazul datelor cu privire la publicare, nu mi se pare deloc necesar să consemnăm numărul de pagini al volumului, formatul sau alte lucruri de acest gen. Datele acestea sunt consemnate doar de către specialiștii în biblioteconomie. Când lucrați la un eseu culegerea unor asemenea date va încărca fișele cu date inutile pentru operațiile de regăsire a informațiilor. Folosiți fișele din bibliotecă, dacă aveți nevoie de asemenea date.

Cea mai delicată este distincția dintre titlu și sursă. Din punct de vedere practic, sursa este ceea ce găsim în fișierul tradițional al bibliotecilor. În cazul unei culegeri, de pildă, vom găsi titlul culegerii și date cu privire la antologia respectivă (cine a alcătuit-o, cum a fost publicată). Nu vom găsi însă fișe pentru fiecare studiu sau fragment din antologie. **titlu vs. sursă**

La fel, în cazul revistelor, în cele mai multe dintre biblioteci, nu vom găsi fișe pentru articole. Vom găsi doar fișe pentru revistele pe care le deține biblioteca. Revista este **sursa** pe care trebuie să o localizăm în cazul articolelor. Inutil am căuta, în cele mai multe cazuri, la autorii respectivi, titlurile de articole.

În fișa bibliografică este, de asemenea, important să introducem o abreviere pentru documentul pe care-l descriem. Abrevierea o vom folosi în fișele de lectură. O putem consemna în același loc ca și-n figura 1.1, dar cu o mențiune din care să reiasă limpede că aceasta este o fișă bibliografică (fișa principală în procesul de descriere a unui document) și nu o fișă de lectură. Există, de asemenea, stiluri de alcătuire a listei bibliografice care folosesc abrevierile.¹⁶ **abrevieri folosite pentru a ne referi la documente**

Este total nerecomandabil să alcătuiți de la început o listă cu descrierile cărților, articolelor etc. pe care le consultați. Lista aceasta, scrisă pe foi sau într-un caiet, este extrem de greu de sortat de îndată ce ia proporții. Fișele, în schimb, pot fi puse cu ușurință în ordinea dorită.

1.2.2.2 Fișe bibliografice pentru studii

Studiile din volumele colective și din publicațiile periodice trebuie neapărat descrise fiecare pe o fișă separată. Este o enormă greșeală să alcătuiți o fișă bibliografică pentru un volum de studii și pe ea să faceți o listă a studiilor din volum. Nu mai puteți manipula informația despre un studiu fără să antrenați și toate celelalte studii în procesul respectiv.

Fișa pentru un studiu seamănă doar până la un punct cu aceea a unei cărți. Există o deosebire fundamentală. Trebuie consemnată **sursa** în care poate fi găsit studiul respectiv.

Să zicem că trebuie să alcătuiți o fișă pentru un studiu dintr-un volum. În mod practic procedați în felul următor. Realizați mai întâi o fișă pentru volum, după indicațiile din §1.2.2.1. Atenție îndeosebi la consemnarea numelui persoanei care a editat volumul. Acest nume

¹⁶V. folosirea etichetelor în fișierele de tip bib la pagina 235, rândul 23.

trebuie urmat de (ed.). Dacă sunt mai multe persoane, puneți pluralul în paranteză.

După ce aveți fișa pentru volum, construiți fișa pentru studiu. Consemnați date cu privire la autoare, autor, respectiv cu privire la text.

trimitere la sursă Blocul de date despre publicare este transformat acum într-o trimitere la fișa volumului, urmată neapărat de paginile între care apare studiul în volum.

În cazul unei reviste, procedăm în același mod, dar în loc să ne referim la un volum includem titlul revistei, volumul, numărul, paginile la care apare articolul, luna sau perioada din an și anul în care a apărut articolul.

Nu este necesar, în cazul revistelor, să facem o fișă cu descrierea bibliografică a revistei. Asemenea date nu se includ în bibliografia eseurilor academice. Alcătuiți o asemenea fișă numai dacă aveți motive speciale să faceți acest lucru.

1.2.2.3 Fișele pentru documentele de pe Internet

Folosiți primele două blocuri de informații din cazul cărților. Diferențele cele mai importante apar la cel de al treilea bloc.

notă despre text În cazul datelor despre text, este util să consemnăm, pe lângă titlu, și alte informații utile: situl pe care se află textul, caracterul textului. Unele documente de pe Internet sunt veritabile cărți electronice, altele sunt adoma unor articole tradiționale, altele sunt simple însemnări. Granițele între tipurile de documente de pe Internet sunt labile; de aceea aceste date trebuie consemnate sub forma unei adnotări nesupuse unor constrângeri formale. Luați în considerare și utilitatea acestor date pentru căutarea pe Internet, în cazul în care adresa are o eroare sau s-a schimbat.

dinamica textelor de pe Internet La faptele legate de publicare, rețineți adresa de pe Internet. Consemnați, dacă găsiți așa ceva, data la care a fost plasat pe Internet documentul. S-ar putea să fie vorba despre un interval de timp. Documentele de pe Internet sunt rescrise din mers. Ele au un caracter **dinamic**. Vechiul concept de „bun de tipar”¹⁷ nu se aplică aici decât foarte rar. Consemnați data la care ați văzut dumneavoastră documentul respectiv pe Internet.

Recomandările formulate aici pentru cărți, studii și documente de pe Internet sunt departe de a acoperi toate cazurile posibile. Din

¹⁷Pentru noțiunea de bun de tipar v. Funeriu[14, p.220].

practică aş putea totuşi spune că ele se potrivesc în marea majoritate a surselor unui eseu filosofic.¹⁸

1.2.3 Fișele de lectură

Studentele şi studenţii care aud pentru prima oară de toate aceste fișe sunt foarte miraţi; uneori sunt chiar revoltaţi de ideea de a folosi fișele. De ce să nu țină un jurnal de lectură? Unii autori recomanda așa ceva.¹⁹

Eu nu sfătuiesc pe nimeni care lucrează la un eseu filosofic să adopte jurnalul de lectură. Ceea ce recomand categoric sunt fișele de lectură. Motivele acestei recomandări pot fi sistematizate în felul următor:

- Jurnalul de lectură s-ar putea să fie folositor în cazul discutării unei opere literare. Pe măsură ce citeşti, notezi fluxul trăirilor pe care le inspiră opera literară. În cazul filosofiei, nu este încă esențial să surprindem trăirile pe care ni le inspiră.
- În al doilea rând, selectarea, sortarea şi regăsirea informațiilor este mult mai ușoară în cazul fișelor. Fișele sunt special concepute pentru a facilita aceste operații.
- În sfârșit, fișele impun o disciplină în consemnarea informațiilor. Nu este întâmplător faptul că ele au dimensiuni standard. Aceste dimensiuni sunt alese în așa fel încât să stimuleze consemnarea ordonată a datelor.

O fișă de lectură are formatul A6 (cartea poștală). Fiecare fișă este divizată în *câmpuri*. Practica proprie ne sugerează că este suficient să împărțim fișa în trei câmpuri.

Primul câmp cuprinde abrevierea care ne trimite la fișa bibliografică adecvată. Abrevierea este bine să fie urmată de indicarea secțiunii din text care stă la baza celor consemnate în fișă. Pot scrie, de

¹⁸Există o serie de surse importante (cum ar fi materialele publicate de către juriști) care nu sunt tratate aici. În măsura posibilului, acest lucru poate fi făcut sub forma unor răspunsuri la întrebări pe situl cărții.

¹⁹Bazerman[2, pp.26–27] recomandă transformarea notelor scrise pe marginea textului într-un jurnal de lectură. Dincolo de pericolul de distrugere a exemplarelor din bibliotecile publice, notațiile marginale conțin și alte primejdii. Sunt în dezacord cu ideea lui Bazerman de a consemna reacțiile afective. În orice caz, acestea nu sunt potrivite cu lucrul la eseu filosofic. Jurnalul de lectură ca atare are un mare dezavantaj față de fișe: căutarea informațiilor este dificilă; sortarea lor după diverse criterii este imposibilă.

Abreviere: descriptor structural	
cuvinte-cheie	Propoziție-cheie

Figura 1.1: Modelul unei fișe de lectură

pildă, MillDL:capitolul:paragraful. Abrevierea trebuie să corespundă cu cea din fișa bibliografică. Descrierea structurală ne permite să nu depindem de paginația ediției consultate. Paginile le vom consemna **doar** în câmpul pentru propoziția-cheie.

Numărul paginii este un fapt legat de modul de publicare a documentului. El nu ne spune nimic în legătură cu structura documentului. De ce este așa de importantă structura documentului ca atare? Dacă avem de comparat ediții diferite sau originalul cu diverse traduceri, paginile nu ne sunt de nici un folos. Trebuie să recurgem la părți, capitole și alte secțiuni, precum și la paragrafe.²⁰ Doar ele ne permit să regăsim cu precizie punctul din text în care apare o anumită idee. Numai textele clasice au o paginație standard la care ne putem referi independent de modul de publicare.

Al doilea câmp cuprinde cuvinte-cheie. De regulă, este vorba despre o pereche de cuvinte-cheie.²¹

Al treilea câmp al fișei de lectură este cel mai important. Aici consemnăm propoziția-cheie pentru secțiunea pe care o adnotăm cu ajutorul fișei de lectură. Este extrem de important să distingem cu **parafr-**
zare vs.
citare mare grijă încă de la acest nivel între parafrizarea unei idei și citat. Eu unul aș recomanda redarea sistematică în propriile cuvinte

²⁰Pentru elementele structurale ale textului v. §1.3.

²¹V. la pagina 22 modul de a analiză a structurii unei propoziții din perspectiva rezumării unui text.

a ideilor-cheie²² și extragerea unui citat doar ca o dovadă a corectitudinii parafrazării operate sau pentru a-l supune unei analize ulterioare.²³

Citatele sunt marcate cu ajutorul ghilimelelor. În cazul în care este vorba de un text în altă limbă, se recomandă consemnarea citatului în original. Traducerea poate fi operată ulterior sau poate fi consemnată într-o paranteză.

Reproduceți citatele fără să le modificați. Când găsiți un element al textului (o eroare de ortografie, de exemplu) care ar putea fi perceput de către publicul cititor ca fiind o greșeală care vă aparține, folosiți *sic*.²⁴ Utilizați însă în mod judicios acest sistem. La urma urmei, până la proba contrarie, se consideră c-ați reprodus în mod corect citatele.

De ce atâta grijă cu citatele? Pentru a evita ceea ce am putea denumi *plagiatul din neglijență*. Dacă nu marcăm adecvat citatele, s-ar putea să le folosim ulterior ca și cum ar fi o parafrază și să nu punem ghilimele. Oricum, chiar și-n cazul parafrăzării, trebuie să menționăm sursa ideii respective.

Pentru a putea preciza ulterior sursa unei idei, în cazul documentelor tradiționale mai avem nevoie de numărul paginii în care am găsit ideea. Un mod comod de a marca pagina este să punem două puncte urmate de numărul paginii (paginilor) în partea dreaptă a câmpului propoziției-cheie.²⁵

Există o mare eroare, pe care am văzut-o comisă de nenumărate ori. Ea constă în consemnarea pe o singură fișă a două sau mai multe propoziții-cheie. Neajunsul acestui mod de a proceda se vede în momentul în care începem să permutăm fișele.

O altă eroare constă în consemnarea în câmpul pentru propoziția-cheie a unor cuvinte-cheie disperate.

În sfârșit, unele persoane, după ce au pregătit fișele, distrug toată

**evitarea
plagiatu-
lui**

**o singură
propozi-
ție-cheie
pe o fișă**

**nu capsăți
fișele**

²²Parafrizarea trebuie făcută cu grijă. Peter Suber observă, într-un text care-l completează pe cel din bibliografia finală, că trebuie să trecem dincolo de „meșterirea superficială a formulărilor” originalului (v. Suber[42, plag.htm]).

²³Prezența citatului pe fișă poate să ne ajute și la realizarea unei parafrăzări corecte. Putem măsura lesne distanța dintre cuvintele originalului și cuvintele noastre. Cum ajungem la „cuvintele-noastre”? Metoda cea mai bună este aceea de a ne constitui pe moment în avocații textului original. Ca avocați trebuie să dăm cea mai bună formă ideilor originalului. În plus, trebuie să ne străduim să fim mai concisi decât originalul.

²⁴Termenul „sic” vine din limba latină și aici înseamnă *așa este scris* în original.

²⁵Nu uitați să notați numerele paginilor. Aveți nevoie de ele atunci când citați!

munca prin capsarea fișelor. În felul acesta ne întoarcem la ideea jurnalului de lectură, respinsă de către noi.²⁶

Textul este ca un *puzzle* cu nenumărate soluții. *Puzzle*-ul ca jucărie are doar o soluție. Textele filosofice nu sunt însă un divertisment. Fișele sunt făcute pentru a reconstitui alte și alte versiuni ale *puzzle*-ului. Dacă le capsăm, blocăm posibilitatea de a studia aceste versiuni.

1.2.4 Fișele de lucru

Ideea introdusă metaforic în titlul primei părți a cărții este cea a unei lumi a ideilor extrem de diverse. Ideile acestea sunt ideile altora, dar și ideile noastre.

consemna- Putem consemna ideile noastre pe fișe asemănătoare cu fișele de
rea lectură. Formatul acestor fișe este însă în mod intenționat restrictiv.
Rolul lor este să ne forțeze să esențializăm.

propriilor În cazul fișelor pentru ideile noastre, scopul nu este esențializarea,
idei ci conturarea unor linii de forță ale argumentelor noastre. Pentru aceasta avem nevoie de loc pentru dezvoltări mai ample.

Formatul adecvat pentru aceste fișe mi se pare A5. Este mai mare decât cartea poștală, dar îți permite să scrii în cele mai diverse locuri. Poți scrie în metrou, folosind o carte, dacă n-aveți un *clipboard* format A5. Poți scrie în parc. Toate aceste locuri se potrivesc perfect cu imperativele „furtunii de idei“.

Nu există reguli precise pentru aceste fișe, așa cum nu există reguli pentru conținutul eseurilor. Nu te poate învăța nimeni ce să pui în conținutul eseului. Trebuie să gândești pe cont propriu.

1.3 În căutarea ideilor-cheie

Câmpul principal al unei fișe de lectură ar trebui să conțină o singură propoziție-cheie. De asemenea, metoda de sortare cea mai bună, în cazul fișelor, este cea care se folosește de cuvintele-cheie din propoziția respectivă.

rezumarea Cum găsim însă cuvintele și propozițiile care reprezintă cheile unui text? Cel mai bun de a învăța acest lucru este acela de a exersa rezumarea de texte. Când rezumăm, de fapt, scoatem la lumină cheile textului. Cu alte cuvinte, producerea de cuvinte-cheie și de

²⁶V. pagina 17 aici.

propoziții-cheie este un efect colateral al operațiunii de rezumare a unui text.

Rezumarea poate fi învățată cel mai bine din practică. Principiile ei de bază sunt relativ simple, dar aplicarea lor cere exercitarea constantă a discernământului uman.

Înainte de a prezenta principiul de bază al rezumării, se cuvine să explicăm felul în care desemnăm elementele din care este compus textul. Elementul de bază, marcat printr-o distanță mai mare între rânduri sau printr-un spațiu alb înainte de primul cuvânt, este denumit curent în românește „alineat”. Există tendința puternică de a numi acest element al textului „paragraf”, după modelul altor limbi.²⁷ Voi adopta o atitudine pragmatică și voi folosi și eu cei doi termeni ca și cum ar avea un înțeles similar.²⁸

„Para-
graf” ca
sinonim
pentru
„alineat”

O altă problemă delicată o ridică termenul de „secțiune”. Din nou, voi extinde în mod pragmatic utilizarea acestui termen. El se va referi, în sens strict, la orice diviziune a textului aflată la un nivel plasat între paragraf (alineat) și capitol. În sens larg, desemnează orice diviziune a textului. Părțile și capitolele reprezintă tot diviziuni ale textului. Ele sunt diviziuni specifice cărților.

secțiune

În general, pentru a nu complica lucrurile, simbolul folosit de noi pentru orice diviziune a textului care are un număr este §.²⁹

simbolul §

Principiul de plecare este că rezumarea începe de la nivelul alineatelor. Alineatele sunt structurile de bază ale textului. Le-am putea asemăna cu moleculele. Ca și moleculele, alineatele au o structură mai fină (atomică și subatomică). Alineatele sunt compuse din propoziții, care au, la rândul lor, o structură.

Gramatica³⁰ pe care o vom folosi poate să pară derutant de simplă, dar este extrem de utilă pentru rezumarea textelor. Fiecare propoziție este desfăcută în două părți: o temă și un comen-

²⁷ Funeriu[14, pp.201–203] discută amănunțit cei doi termeni. Pentru opțiunea în favoarea sinonimiei v. și Moroșanu[30, pp.17, 55].

²⁸ Folosit cu un înțeles distinct, „paragraf” înseamnă *cea mai mică secțiune de text formată din unul sau mai multe alineate*. Iancu și Horvat, de pildă, numerează paragrafele înțelese în sensul de mai sus. V. explicația din Iancu și Horvat[19, p.10].

²⁹ Comanda pentru obținerea în L^AT_EX a acestui simbol este `\textsection{}` și are, nu întâmplător, un nume foarte sugestiv.

³⁰ Strict vorbind, este vorba doar despre o idee din gramatica sistemică. Acest tip de gramatică distinge între diverse sisteme funcționale ale unei limbi. Unul dintre aceste sisteme este cel analizabil în termenii distincției dintre *temă* și *remă*. A se vedea reconstrucția acestui gen de gramatică în cartea lui Terry Winograd, *Language as a Cognitive Process* (Reading: Addison-Wesley, 1983), pp.272–310, dar în special pp.282–285.

tariu.³¹ Comentariul este ceea ce are de spus autoarea sau autorul pe tema respectivă. Într-un alineat, o temă poate fi comentată în mod repetat. De asemenea, un comentariu poate să devină tema unei alte propoziții.

temă vs. comentariu Fiecare alineat are o temă-cheie și un comentariu-cheie pe tema respectivă. Ele formează împreună propoziția-cheie a alineatului respectiv sau, mai simplu spus, cheia alineatului.

Cheia alineatului poate să apară explicit sau să fie implicită. În cazul în care este implicită, ea trebuie reconstituită. Adevărul este că obiceiul de a folosi chei implicite face textul obscur. Când scriem ar trebui să ne ferim de cheile implicite. Există însă mari autori pentru care merită să facem efortul de a căuta cheia implicită.

tipuri de alineate Evident, alineatul ideal n-are decât o singură cheie. Dacă ea este pusă la începutul alineatului, atunci alineatul se poate spune că are o structură descendentă.³² Dacă este plasată la mijloc, alineatul este pivotant. Dacă este la sfârșit, alineatul are o structură ascendentă.³³

Fiecare autor are stilul său și, dacă analizăm atent un număr suficient de mare de alineate, ne vom da seama unde plasează cheile. Acesta este un pas important în cazul în care vrem să parcurgem foarte rapid textul.

împarantezarea Mulți autori folosesc împarantezarea: alineatul se deschide cu ideea-cheie și se încheie cu o remarcă privind consecințele acesteia. Între aceste două elemente se află partea care susține, dezvoltă sau ilustrează ideea-cheie. Propoziția finală poate consta și dintr-o simplă reafirmare a ideii-cheie.³⁴

Fiecare secțiune este compusă dintr-un număr de alineate (paragrafe). Dacă extragem cheile acestor alineate, atunci putem analiza legăturile dintre ele și putem desprinde cheia la nivelul fiecărei secțiuni.

Dacă este vorba despre o carte, de la nivelul secțiunilor trecem

³¹Ideea de a folosi distincția temă-remă în procesul de rezumare este folosită de către Dubovik, Veize și Golovko[10]. De-a lungul anilor am folosit în practică această tehnică și ni s-a părut utilă. Terminologia folosită aici este una adaptată, după cum se va vedea și-n continuare.

³²În manualul autorilor din Bielorusia[10, p.13] acest tip de alineat este denumit „alineat deductiv“. Cred că denumirea induce în eroare. S-ar putea ca ideea-cheie să se sprijine pe o inducție, dar autorul să o plaseze la începutul alineatului.

³³În manualul lingviștilor din Bielorusia[10, p.14] denumirea este de „alineat inductiv“.

³⁴Răspândirea acestui tip de alineat n-ar trebui să ne mire. El este recomandat, folosind este drept o altă terminologie, în influentul manual al lui Strunk[41, p.15].

apoi la nivelul capitolelor și al întregului text. Rezumatul nu face decât să reconstituie structura relațiilor dintre cheile diviziunilor textului.

Acum se vede foarte limpede care este rolul fișelor de lectură. În cazul unei lecturi lente, fișele de lectură surprind cheia fiecărui alineat. Pot fi făcute fișe chiar până la nivelul propozițiilor, dar numai în cazul unei lecturi deosebit de amănunțite. Dacă pregătim, de pildă, o traducere, atunci este normal să procedăm așa. Ideal ar fi chiar să analizăm în detaliu fiecare propoziție și să consemnăm atent legăturile cu restul textului. **lectura lentă**

În cazul lecturii ceva mai rapide, sunt consemnate doar cheile secțiunilor. Lectura foarte rapidă a unei cărți nu surprinde decât cheile capitolelor sau poate doar o intuiție cu privire la ideea-cheie a întregii lucrări. **lectura rapidă**

Cuvintele-cheie sunt elementele (cuvinte sau sintagme) în jurul cărora gravitează temele sau comentariile. De aici și ideea de a avea, de regulă, o pereche de cuvinte-cheie. **cuvinte-cheie**

Utilitatea cuvintelor-cheie este foarte mare. Extragerea lor, ca să spunem așa, trebuie făcută **manual**. Doar persoana umană, care înțelege textul, poate să identifice elementul crucial al unei teme sau al unui comentariu.

De asemenea, nu trebuie să uităm să surprindem ideea-cheie a întregului text. Cheile nu trebuie înșiruite în funcție de numărul de ordine al capitolelor. Accentul trebuie să cadă pe dezvăluirea legăturilor dintre capitolele și părțile lucrării. Într-o lucrare de filosofie ne așteptăm la legături logice între alineate și între secțiuni. Nu toate textele de filosofie reușesc să aibă o arhitectură argumentativă unitară. Există însă, mai mult sau mai puțin explicit, o rețea de argumente. Această rețea este străbătută de firul ideii-conducătoare a întregului text.

1.3.1 Reconstituirea argumentelor

Identificarea, analizarea și reconstrucția argumentelor nu constituie sub nici o formă obiectul cărții de față. Trebuie consultat un text de specialitate pentru această importantă operație intelectuală.

Ceea ce ne interesează aici este impactul muncii de reconstituire a argumentelor asupra lucrului cu fișele.

Premisele și concluzia, precum și alte elemente aferente unui argument (definiții ale unor concepte, presupoziii care afectează argumentul) pot fi consemnate pe fișe. Trebuie să verificăm însă dacă

secvența formată din abreviere și descrierea structurală este suficient de scurtă. Comprimăm eventual doar descrierea structurală, aducând-o la forma unui scurt șir de numere.

**schema
argumen-
tului** Când abrevierea este pregătită în modul descris mai sus putem să folosim o fișă de lucru pentru a construi schema argumentului.³⁵ Ne putem sluji de un manual de argumentare pentru a găsi sugestii referitoare la reprezentarea relațiilor dintre elementele raționamentelor din text. Dacă nu vă plac schemele simbolice, aveți totuși nevoie de un ghid care să vă orienteze în mod serios în domeniul argumentării.³⁶

Dacă argumentul are o structură mai simplă o fișă tip carte poștală s-ar putea să fie suficientă pentru schema argumentului. Doar abrevierile baroce ar putea crea probleme.

Dacă vrem să evaluăm argumentul sau să schițăm diverse idei pe marginea lui, atunci trebuie să recurgem la fișele de lucru.

De asemenea, este important ca, în final, să integrăm argumentul reconstituit în rezumatul pe care l-am făcut după metoda indicată mai sus.

1.4 Revoluția computerelor și munca intelectuală

Impulsul de a scrie această carte în forma actuală a venit în timp, pe măsură ce constatam că tot mai multă lume utilizează calculatorul într-un mod care ne întoarce într-o epocă de sălbăticie intelectuală. Lipsesc semnele diacritice la literele românești, secțiunile nu au numere, bibliografiile nu există, nu se alcătuiesc indici. Se investește mult în partea fizică a computerelor, dar extrem de puțin în învățarea programării. Când se învață câte ceva despre partea subtilă a computerelor, din păcate, chiar și la nivel universitar, totul se limitează la programele comerciale de birotică.

Există o soluție la problema de mai sus: adoptarea limbajului folosit pentru tehnoredactarea textelor matematice. Deviza de pe frontispiciul Academiei din Atena, ἀγεωμέτρητος μηδεὶς εἰσὶτω,³⁷

³⁵ Pentru construcția schemei se pot folosi săgeți etichetate, de genul celor din rețelele semantice. Etichetele sugerează relația dintre propozițiile-cheie. Putem arăta astfel pe care propoziții se sprijină o concluzie mai importantă.

³⁶ Cartea lui Flew[13] este un exemplu excelent de astfel de ghid care nu recurge nici măcar la o singură formulă. Ea ne arată în schimb cum pot fi puse sub lupa logicii argumentele reale.

³⁷ Expresia figurează la cel puțin nouă autori vechi. Între alții, o folosește Da-

ar trebui schimbată și referirea la cunoașterea geometriei înlocuită cu cerința de a cunoaște limbajul $\text{T}_{\text{E}}\text{X}$ ³⁸ sau sistemul $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$,³⁹ bazat pe $\text{T}_{\text{E}}\text{X}$.

1.4.1 Dilema muncii intelectuale computerizate

Cele mai multe persoane învață relativ ușor să creeze un fișier, dar nici nu se gândesc să învețe cum să structureze fișiere de tip text. Ele își procură, pe căi mai mult sau mai puțin legale, un program care are în numele său cuvântul magic *office*. Cu acest program se apucă să elaboreze manuscrise.⁴⁰

Elaborarea manuscrisului implică însă un proces de rescriere a textului. Un eseu academic are însă secțiuni numerotate, note complicate, trimiteri interne. Dacă introducem o secțiune nouă, atunci toată numerotarea se schimbă. Trimiterile interne vor trebui modificate. Elementele care dau nota distinctivă a unui eseu academic sunt *fragile*. Noile versiuni pot să piardă ceea ce fusese câștigat anterior.

Cum se descurcau oamenii în mod tradițional? Lucrul acesta se vede și dintr-o privire superficială asupra termenului de *manuscris*. Etimologia sa este cât se poate de transparentă: este vorba despre un text scris cu mâna. Toate operațiile implicate de trecerea la o nouă versiune se făceau manual. Când totul era gata manuscrisul era dactilografiat.

Tradițional, marea majoritate a lucrărilor de seminar erau prezentate ca manuscrise. Lucrările de diplomă și cele de doctorat se dactilografiau.

Computerul permite acum producerea unor texte care au calitatea celor tipărite. Este destul de evident pentru oricine că un program de calculator adecvat transformă computerul în ceva ce este mult mai mult decât vechea mașină de scris.

vid (filosof armean din secolul al VI-lea). Gabriel Liiceanu o traduce, cu acuratețe, prin „nici un necunosător al geometriei să nu intre”, în David, *Introducere în filosofie* (București: Editura Academiei, 1977), p.74. V. textul grecesc în A. Busse(ed.), *Commentaria in Aristotelem Graeca* (Berlin: Reimer, 1904), p.57, rândurile 19-20. David socotește că pentru Platon matematica era un exercițiu pregătitor pentru filosofie.

³⁸Creatorul $\text{T}_{\text{E}}\text{X}$ este profesorul Donald Knuth de la Stanford University. V. pagina sa de web la adresa <<http://www-cs-faculty.stanford.edu/~knuth/>>. Pentru mai multe informații despre Donald Knuth v. aici pagina 183.

³⁹ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, v. aici anexa B, este creat de către Leslie Lamport. În prezent Lamport este cercetător la Microsoft. V. pagina sa de web la adresa <<http://research.microsoft.com/users/lamport/>>.

⁴⁰Cărțile lui Funeriu[14, passim] sau Biriș[3, pp.141-144] sunt total sub vraja *office*-urilor.

Alegerea între programele de calculator ne pune însă în fața unei dileme. Ce fel de programe să alegem?

- unele pentru care ne trebuie „permis de utilizator“?
- sau unele pentru care ne trebuie „permis de programator“?

Pentru imensa majoritate a intelectualilor de factură umanistă chiar și utilizarea pare să fie un obstacol major. Ei sunt tentați de farmecele sistemelor WYSIWYG.⁴¹

Printre filosofi există evident o excepție. Este vorba despre persoanele care au fost atrase de logica simbolică, sub o formă sau alta. Pentru respectivele persoane nu este deloc greu să se îndrepte către programare.

Pledoaria din această carte este desigur pentru adoptarea, în munca intelectuală, de către toată lumea a perspectivei programării. Am putea insista asupra ideii de **perspectivă**. Am putea spune că nu este vorba despre scrierea nici unei linii de program. În fapt, chiar așa și stau lucrurile dacă folosim șabloane gata pregătite. Credem însă că n-are rost însă să vindem iluzii.

Perspectiva programării cere efectiv să înțelegi, fie și la un nivel foarte elementar, ce anume înseamnă un program. S-ar putea ca lucrul acesta să fie chiar mai simplu de învățat decât utilizarea avansată a unui program. Problema este însă cu totul alta. La ce ajungem dacă nu ne orientăm către perspectiva programării?

Adevăratul dramatism al dilemei devine vizibil dacă răspundem la întrebarea de mai sus. În munca intelectuală sunt folosite unelte din pachete care au în numele lor cuvântul englezesc *office*. Acest cuvânt înseamnă „birou“ și ne furnizează cheia dilemei. Pachetele respective sunt adecvate pentru munca de secretariat. Ele nu sunt concepute pentru a fi utilizate în munca intelectuală.

Ce anume face ca pachetele destinate muncii de birou să fie nepotrivite pentru munca intelectuală? Personalul din secretariate nu trebuie să aibă o atitudine creatoare față de conținut; ar fi și absurd să facă așa ceva. Menirea sa este cu totul alta.

**separarea
conținutului de
formă**

Munca intelectuală presupune însă independență și o atitudine creatoare față de conținut. Cea mai modestă lucrare de seminar n-are voie să fie un simplu rezumat sau un colaj de rezumate. Trebuie să existe un mod propriu de a prezenta ideile altora.

⁴¹ Aceste sisteme sunt bazate pe ideea că vezi pe ecran textul așa cum va fi el tipărit.

Pentru a putea să ne concentrăm asupra conținutului trebuie să existe o separare între conținut și formă. Exact lucrul acesta nu-l fac programele uzuale pentru munca de birou.

În al doilea rând, oricât ar părea de curios, programul trebuie să imprime o disciplină muncii intelectuale. Din nou, programele pentru munca de birou nu sunt orientate către imprimarea disciplinei necesare muncii intelectuale.

Ce înseamnă această disciplină? Ea începe, de pildă, chiar cu modul în care construim paragrafele textului nostru. Pot fi oferite multe alte exemple. Un program destinat muncii de birou nu ne va forța să avem toate elementele obligatorii într-o trimitere bibliografică. În fapt, nu ne va forța să folosim un stil uniform în bibliografia noastră.

În al treilea rând, în practică, uneltele programatorilor permit un management superior al notelor de subsol, trimiterilor și indicilor. Paginile documentului pregătit pentru tipărire sunt, de asemenea, stabile atunci când sunt pregătite prin programare. Nu riști ca pe altă mașină sau la o nouă rulare a programului să vezi altă aranjare a paginilor.

În rezumat, dilema este între folosirea unor programe seducătoare, care te îmbie să le utilizezi, dar nu sunt adecvate muncii intelectuale, și o abordare considerabil mai abstractă, dar potrivită cu munca intelectuală.

1.4.2 Investiți într-un computer

Nu spuneți că nu vă puteți permite să achiziționați un computer. Evident, totul pe lumea aceasta are un cost. Un cost înseamnă însă jertfirea unor oportunități de acțiune. Sunt destule lucruri pe care merită să le jertfiți pentru a putea lua un computer și sistemul de operare aferent.

Un computer costă acum cât apometrele dintr-un apartament. La mine în cartier, mai toate apartamentele au apometre. N-am idee în câte apartamente există un computer, dar cred că acest număr nu se compară cu acela al apometrelor. În orice caz, puțini dintre cei care au conexiune la televiziunea prin cablu au adăugat și o conexiune Internet.

Persoanele care spun că nu-și pot permite să ia un computer se gândesc la „ultimul model“, cu DVD și placă grafică pentru jocuri fabuloase. Cred că este o adevărată crimă să folosești computerul pentru a te uita la DVD și a te distra cu jocuri de calculator. În fond, există DVD separat și stații speciale pentru jocuri. Compu-

terul este cu adevărat util pentru aplicațiile subtile din punct de vedere intelectual.

Prețul unui sistem pe care ar putea fi folosite elementele esențiale ale tehnicilor care vor fi explicate aici este efectiv modic. Ar putea fi un computer achiziționat la mâna a doua și pe care să poată fi scrise fișierele `LATEX` care vor fi transferate pe o dischetă și compilate în laboratorul Facultății.⁴²

Achiziționați computerul de la o firmă care vă oferă consultanță. Puteți astfel potrivi bugetul dumneavoastră cu echipamentul cumpărat.

Când este vorba despre computere mai există o serie de alte obstacole în calea utilizării lor în munca intelectuală. În laboratorul Facultății predomină computerele pe care sunt instalate sisteme de operare Unix. Când utilizatorul obișnuit cumpără la magazin un computer, de regulă, achiziționează un PC cu Windows preinstalat. Tipul acesta de computer predomină, de altfel, și-n catedrele Facultății.

**ucenici
vrăjitori** Trebuie să facem unele distincții între persoanele care utilizează computere și să explicăm apoi cui vrem să ne adresăm. Marea masă este compusă din utilizatori obișnuiți. Alt tip de utilizator caută să aibă un computer cât mai puternic. Acesta va fi tentat de ultimul tip de sistem Windows și de tot felul de programe care sunt, precum motocicletele sau mașinile sport, menite să dea senzația unei utilizări în forță a computerului. O altă categorie o constituie cei pe care-i vom denumi **ucenici vrăjitori**.⁴³ Aceștia instalează Linux pe computerul lor. Folosesc un editor pentru programatori. Compilează programe care sunt scrise în C/C++. Chiar scriu asemenea programe.

anexele Anexele cărții noastre se adresează utilizatorilor obișnuiți. Îi ignoră pe cei care utilizează computerul în forță. Unele porțiuni cuprind idei pentru ucenicii vrăjitori, dar nu o prezentare în detaliu a informațiilor respective. Ucenicii vrăjitori vor experimenta, dacă sunt curioși, în zona respectivă. De asemenea, utilizatorii obișnuiți trebuie să știe că experimentele respective sunt mai mult sau mai puțin riscante. Dacă iau în mână bagheta de ucenic vrăjitor o fac pe propria răspundere.

În practică, am testat toate programele despre care este vorba în carte sub Windows98SE și, în punctele cele mai importante, și sub WindowsXP. După părerea noastră, Windows98SE este un sistem

⁴²Pentru detalii privind acest mod de lucru a se vedea aici §B.1.2.2.

⁴³Am preluat expresia din cartea lui Raymond Seroul[39].

foarte potrivit pentru un computer care nu este conectat la Internet și pe care principala activitate este scrierea de lucrări de seminar, teze, disertații, articole, cărți. Pentru investigațiile pe Internet am folosit un client Windows98SE al unui server Linux și serverul Linux ca atare. În laboratorul Facultății studențele și studenții au acces la un server Linux similar, precum și la un sistem Sun Solaris, care este tot de tip Unix.

Filosofiile care stau în spatele unui sistem Unix și al unuia de tip Windows sunt cât se poate de diferite. Un sistem Unix este ca o trusă cu sute, chiar mii de unelte. Cu fiecare unealtă se poate face doar un anumit lucru. Se lucrează ca pe o bandă rulantă: modifici obiectul prelucrat cu o unealtă; îl pui pe bandă; apoi se execută o nouă operație de prelucrare și așa mai departe. În sistemul Windows ideea este de a crea o mașină integrată de mari dimensiuni care să facă ea toate operațiile de mai sus. Integrarea simplifică enorm utilizarea, căci mașinăria pornește prin apăsarea unui buton virtual pe ecran. Reglarea mașinii integrate se face printr-o interfață grafică, ceea ce este efectiv agreabil din perspectiva utilizatorului. Pe măsură însă ce gradul de integrare crește, interfața devine tot mai complexă și trebuie să consulți cărți groase pentru a ști ce și cum să potrivești.

1.5 Computerul pregătit pentru munca intelectuală

Fiecare capitol din trunchiul principal al cărții de față se încheie cu o scurtă secțiune care face legătura cu anexele.

Dacă vă hotărâți să folosiți computerul în munca intelectuală, nu abandonați total fișele tradiționale. Trecerea de la metodele tradiționale ale muncii intelectuale la cele actuale nu a fost, de altfel, una bruscă.

Unul dintre pașii importanți în procesul de tranziție l-a reprezentat trecerea de la sortarea manuală a fișelor la cea mecanică. În unele situații, cum ar fi prelucrarea recensămintelor de mare amplitudine, sortarea manuală devenise extrem de inefficientă la sfârșitul secolului al XIX-lea. Hermann Hollerith, statisticianul care a inventat în 1887 mașina de sortat cartele perforate, a rezolvat această problemă.⁴⁴

Hermann Hollerith a fost, de altfel, și fondatorul Tabulating Ma-

⁴⁴Arnold, Hill și Nichols[1, pp.41-43] descriu pe larg impactul invenției lui Hollerith.

chine Company.⁴⁵ În 1911, firma înființată de Hollerith, în urma unei fuziuni, a devenit IBM. IBM s-a transformat mai târziu într-un mare producător de calculatoare. Tot IBM este firma care, la începutul anilor '80 ai secolului trecut, a promovat pe piață PC-urile.

În anii '70 ai secolului trecut, computerele nu erau accesibile decât unui număr restrâns de persoane. Contactul masiv cu calculatoarele personale a început abia în anii '80. La începutul anilor '90 rețelele de calculatoare erau încă frave. Doar către sfârșitul decădei Internetul capătă o importanță crucială. Din perspectiva muncii academice, Internetul devine o bibliotecă planetară.

În cazul în care trebuie să o porniți de la zero, citiți mai întâi anexa A. Încercați să înțelegeți bine ideea de fișier. Pentru a putea înlocui fișele bibliografice tradiționale cu fișe electronice studiați anexa B.2.2.

La o primă lectură puteți sări peste anexa despre expresiile regulate (§A.3) și aspectele mai avansate ale căutării în fișierele computerelor (§A.4). Reveniți la ele după ce citiți capitolul despre importanța proiectului de cercetare (§2).

⁴⁵ Arnold, Hill și Nichols[1, p.43] arată că Hollerith a înființat firma sa pentru comercializa mașinile de sortat cartele.

Capitolul 2

La început a fost proiectul

Cuprins

2.1	Proiectul de cercetare	32
2.1.1	Tema proiectului	33
2.1.2	Bibliografia inițială	33
2.1.3	Obiectivele proiectului	34
2.1.4	Ipoteza proiectului	35
2.2	Documentarea	36
2.2.1	În căutarea bibliografiei	36
2.3	Publicul cititor	39
2.3.1	Ochiul critic al colegilor și colegelor . .	41
2.4	Construiți proiectul cu ajutorul computerului .	41

Cum începem lucrul la eseu nostru? Răspunsul este foarte simplu: ne documentăm și cercetăm. Mai precis formulat, principiul avut în vedere ar suna astfel: n-are sens să vă faceți un „plan cu ideile“ pe care le va conține eseu final până n-ați investigat problema care vă interesează.¹

Principiul conform căruia „la început este o cercetare“ este foarte bun pentru a para o eroare extrem de frecventă. Cel puțin până acum, majoritatea studentelor sau studenților care au vrut să fie conducătorul lucrării lor de diplomă au avut tendința de a-mi arăta, din primul moment, un plan al lucrării finale.

Prezinti un plan al lucrării tale unui editor. Eu nu sunt însă un editor sau un redactor al lucrărilor studentești. Ceea ce ar trebui să fac este să îndrum o investigație și să ofer sfaturi pe parcursul elaborării lucrării. Oricât de pompos ar suna, trebuie să ajut oamenii să gândească.

¹Planul vine la urmă, când asamblați secțiunile textului! V. aici pagina 50.

2. La început a fost proiectul

Mai rău chiar, cei mai mulți păreau destul de nedumeriți când le ceream amănunte cu privire la țelurile investigației lor.

Constrângerile impuse în continuare proiectelor s-ar putea să nu fie pe gustul unora. Împiedică ele oare manifestarea inteligenței proprii? Nu.

Pe de o parte, ar fi un non-sens evident ca o lucrare de seminar să-și propună din start acoperirea unei arii tematice prea vaste. Chiar dacă lucrul acesta este mai puțin evident în cazul lucrărilor de diplomă, disertațiilor sau tezelor de nivel mai avansat, circumscrierea cu grijă a temelor este și acolo un lucru important.

Pe de altă parte, trebuie acordată atenție unui moment asupra căruia am accentuat mai puțin: analiza argumentelor filosofice. Acest lucru poate fi făcut la diverse niveluri de sofisticare și nu există aici posibilitatea de a trage o graniță netă între lucrările de seminar și contribuții găzduite de publicații profesionale de nivel înalt.

Ceea ce distinge proiectul de cercetare în vederea alcătuirii unei lucrări de seminar este prudența cu care în mod normal ar trebui formulate obiectivele. Din experiență am învățat că este cel mai greu să-i convingi pe studenți și pe studente că obiectivul general al oricărei lucrări studențești este acela de a arăta că ai înțeles o literatură și o poți prezenta cu mijloace proprii, fără a fi sclavul sau sclava formulărilor din texte investigate.

Prin contrast, dacă vrei să pătrunzi într-o revistă cu pretenții ridicate este necesar să încerci să arăți ceva mai mult decât faptul că ești familiarizat cu o dezbatere din literatura de specialitate. Ar trebui să aduci o contribuție cât de mică într-o problemă la care ai meditat și alții. Trebuie să ai un *rezultat*.

2.1 Proiectul de cercetare

Proiectul cu care plecăm la drum este așternut pe o simplă foaie de hârtie. El cuprinde titlul proiectului, identificarea unei discuții din literatura de specialitate (cu ajutorul a două-trei trimiteri bibliografice), obiectivele proiectului și o ipoteză de lucru.

Schematic elementele proiectului arată în felul următor:

- tema proiectului
- o bibliografie minimală
- obiectivele proiectului și ipoteza de bază a proiectului

2.1.1 Tema proiectului

Titlul proiectului trebuie să sugereze, la lucrările academice, cât mai bine tema. O foarte scurtă prezentare nu este neapărat necesară. Ar putea fi însă utilă. În câteva cuvinte vor fi schițate reperele de bază ale temei propuse.

Cum să fie aleasă tema? Există unele tehnici relativ sofisticate de găsim a temei unei lucrări. Modul însă în care ar trebui aleasă tema unei lucrări de seminar ar trebui să fie relativ simplu.

O lucrare de seminar reprezintă proba unei munci independente de tip elementar. La curs nu pot fi discutate în profunzime toate conceptele sau argumentele care sunt întâlnite într-un domeniu. La seminar nu pot fi analizate și dezbătute toate textele semnificative. Prin lucrarea de seminar sau eseul prezentat la examen, studenta sau studentul dovedesc capacitatea de a studia în mod independent concepte sau argumente din bibliografia adecvată pentru cursul și seminarul respectiv.

De exemplu, cursul ar putea să prezinte aspecte ale operei lui Friedrich von Hayek, dar să nu insiste asupra concepției sale despre ceea ce în engleză se numește *rule of law* (supremația dreptului). Aceasta ar putea fi tema unui eseu. **teme acceptabile**

Un alt tip de temă de eseu ar de genul „ideea de supremație a dreptului la A.V.Dicey și F.A.Hayek“. Comparațiile de acest gen sunt o sursă inepuizabilă de teme.²

Teme de genul „sinuciderea“, „avortul“, „pedeapsa cu moartea“ ar trebui să nu fie acceptate. De câte ori am fost de acord cu asemenea teme, am regretat apoi acest lucru. Acestea sunt teme prea vaste pentru o lucrare de seminar sau o lucrare de diplomă. De asemenea, obiectivele proiectului nu au cum să fie clare. Temele acestea trebuie refăcute în procesul de rescriere a proiectului. **teme inacceptabile**

În cel mai bun caz, teme de genul de mai sus repetă ceea ce s-a făcut în curs. Obiectivele unui curs sunt prin natura lor diferite de cele ale unei lucrări de seminar. Iar o lucrare de diplomă nu este un curs sau o carte.

2.1.2 Bibliografia inițială

Bibliografia proiectului este doar o mică parte a bibliografiei pe care o vom alcătui pe parcursul investigației. Ea are un rol foarte precis: identificarea unei dezbateri în orizontul temei date.

²Pentru eseuri care discută un singur autor v. ghidul lui Peter Suber[42].

2. La început a fost proiectul

Alegerea cu grijă a bibliografiei inițiale are un rol capital în dezvoltarea în continuare a proiectului. Există două condiții minimale ale acestei bibliografii.

Prima este aceea ca sursele bibliografice să fie disponibile efectiv. Cu cât le putem consulta mai ușor și mai des cu atât mai bine.

A doua condiție este ca minimum una dintre surse să aibă o dată cât mai recentă și trimiteri cât mai numeroase la literatura de specialitate. Chiar dacă discutăm un autor foarte vechi, cum ar fi Platon, trebuie să avem la îndemână comentarii cât mai noi.³

2.1.3 Obiectivele proiectului

Când am identificat o dezbatere putem trece la faza următoare a construcției proiectului: stabilirea obiectivelor. Principiul fundamental este aici acela că ne alăturăm unei dezbatere și încercăm să participăm la ea.

Obiectivele sunt elementele esențiale ale proiectului. Ele reflectă intențiile cercetării. În principiu, obiectivele sunt de forma „încerc să răspund la întrebarea ...”. Altfel spus, obiectivele arată că vrem să încercăm să rezolvăm o anumită problemă.

Obiectivele nu trebuie să fie numeroase. La o lucrare de diplomă, două-trei obiective sunt absolut suficiente. O singură persoană, în câteva luni, nu poate să atingă prea multe țeluri. În cazul unei lucrări de seminar, ar trebui să existe un singur obiectiv și numai unul.

Nu există sfaturi valabile în toate situațiile sau liste generale de subiecte pentru lucrări. Fiecare proiect trebuie discutat separat. Tema, bibliografia sau obiectivele trebuie adaptate la capacitățile și interesele de cunoaștere ale persoanei care va elabora lucrarea.

Am putea identifica oare o problemă pe care nu a mai discutat-o nimeni până acum? În principiu, acest lucru este desigur posibil. Dar încercarea de a rezolva problema respectivă nu poate fi nicicum un obiectiv al unui exercițiu academic, inclusiv la nivelul unei teze de doctorat. Obiectivele de acest tip sunt, prin natura lor, situate într-o nișă care nu permite comparația cu rezultatele obținute de alții.

³ Luați ca model o carte precum cea a lui Valentin Mureșan, *Comentariu la Republica lui Platon* (București: Metropol, 2000). La paginile 293–294 găsiți o listă impresionantă de comentarii recente ale textului lui Platon. Autorul a separat chiar el aceste surse de bibliografia generală, mult mai amplă.

Adevărul este că și revistele de specialitate serioase nu publică, de regulă, studii care reflectă asemenea retrageri într-o nișă.

Chiar atunci când publică articole care lansează o nouă dezbatere, revistele de prestigiu nu se angajează, de fapt, într-o nișă. Articolele respective pot combina, de pildă, elemente ale unor linii de cercetare deja existente. Ele pot, de asemenea, critica în mod radical un rezultat standard și introduce astfel o nouă problematică.

În cazul exercițiilor academice, nici acest tip de proiecte nu este rezonabil. Motivul principal îl reprezintă imposibilitatea de a evalua rezultatele obținute. Evaluarea unor asemenea contribuții se face doar în timp, printr-o intensă discuție critică.

2.1.4 Ipoteza proiectului

Ca și tema sau obiectivele proiectului, ipoteza, în cazul unei lucrări studențești, nu trebuie să aibă ambiția de a reprezenta un *rezultat* în sensul tare al cuvântului.

Tipică este situația în care comparăm doi autori. S-ar putea ca toată discuția din literatura de specialitate să se învârtă în jurul convergențelor și divergențelor între teoriile celor doi. Ipoteza ar putea fi foarte simplă: „au dreptate cei care pun accent pe convergențe” sau „au dreptate cei care pun accent pe divergențe”.

Condiția fundamentală la ipoteze de genul celor de mai sus este să nu le formulați „dialectic”: au și unii, au și alții dreptate. Așa ceva n-are sens din punct de vedere logic. Puteți distinge doar între planuri în care există convergență și planuri în care predomină divergențele.

Ipoteza proiectului este testată pe măsură ce lucrăm la proiect. Există cel puțin două modele ale dezvoltării unui proiect.

Unul este modelul în cascadă. În acest model ipoteza este menținută constantă. Sunt dezvoltate treptat diversele componente ale textului, pe măsură ce sunt atinse obiectivele proiectului.

Alt model este cel în spirală. Acest model implică ajustarea și modificarea ipotezei de bază, atâta timp cât textul nu este intrat în faza finală a elaborării sale. Textul este scris și rescris; sunt operate numeroase rearanjări ale elementelor sale.

Eu unul înclin categoric către al doilea model.

**dezvolta-
rea
proiectu-
lui**

2.2 Documentarea

Documentarea începe încă din momentul în care construim proiectul eseului. Ea ne permite să reperăm literatura relevantă în orizontul temei pe care am vrea să o cercetăm.

De asemenea, documentarea ne permite să precizăm tema investigației. Documentarea preliminară trebuie făcută imediat după ce avem intuiția că o anumită temă ar ascunde probleme interesante.

trei pachete cu fișe Rezultatele documentării iau forma unor pachete cu fișe bibliografice, a unor pachete cu fișe de lectură și a pachetelor cu fișe în care am reconstituit argumentele din textele investigate. Tehnicile menționate în secțiunea despre rezumare⁴ le folosim pentru a desprinde propozițiile-cheie și legăturile dintre ele.

corelația cu fișele de lucru Scopul eseurilor nu este acela de realiza o lucrare cu caracter documentar. De aceea n-are sens să dezvoltăm pe baza fișelor de lectură rezumate de sine stătătoare ale textelor. Fișele de lectură trebuie corelate în timp cu fișele de lucru pe care notăm propriile noastre idei. A îngheța totul sub forma unor rezumate masive este o acțiune contraproductivă.

2.2.1 În căutarea bibliografiei

De îndată ce proiectul este stabil, se poate trece la parcurgerea literaturii relevante din perspectiva obiectivelor pe care și le propune proiectul. Aici se vede imediat de ce este important să avem niște obiective. Fără ele am aduna munți de date bibliografice, fără să știm dacă ne folosesc sau nu efectiv.

Există mai multe metode de construire a bibliografiei. Pe cele mai importante le vom discuta în continuare.

2.2.1.1 Fișierul tematic tradițional

În viitor, s-ar putea să vă întâlniți, în bibliotecă,⁵ tot mai rar cu fișierul tematic tradițional. În genere, fișele tradiționale din hârtie tind să fie înlocuite cu sisteme computerizate. Principiul construirii acestor fișiere tematice a fost discutat atunci când am vorbit despre sistemele de clasificare.⁶

⁴V. §1.3 pentru tehnicile rezumării.

⁵Pentru a descoperi rețeaua bibliotecilor și librăriilor folosiți un repertoriu special, cum ar fi, de exemplu, cel realizat de CIMEC și MERONIA, *Romania Culture Directory 2003*(București: MERONIA, 2002), pp.57-87, 96-113.

⁶V. §1.1.3 pentru sistemele de clasificare a surselor în biblioteci.

Un fișier tematic, după cum sugerează și numele său, grupează sursele nu după numele autorilor, ci după codurile din clasificarea folosită de către bibliotecă.

În practică, trebuie să găsiți un fișier cu teme și codurile aferente lor. Este, de asemenea, recomandabil să consultați personalul de specialitate al bibliotecii, care vă va da lămuriri cu privire la codurile care vă interesează. De îndată ce știți codurile puteți trece la identificarea surselor care sunt relevante din perspectiva obiectivelor proiectului de cercetare.

**fișierul cu
codurile
CZU**

Fișierul tematic este cât se poate de util. Marea lui problemă la noi era felul în care era structurat un asemenea fișier înainte de 1989. Conform modului de gândire totalitar, existau o serie de opinii infailibile. Totul era clasificat în funcție de concordanța sau discordanța cu aceste susțineri infailibile. Concepția aceasta nu este doar îndoielnică din punct de vedere filosofic, ci conduce și la deformări monstruoase ale fișierelor tematice. Chiar și codurile CZU au fost modificate pentru a pune clasificarea în acord cu acest mod de a vedea lucrurile.

Refacerea tuturor fișierelor tematice din biblioteci este o muncă fără îndoială colosală. Nu este de mirare că ea nu a fost întreprinsă.

Morala este că nu trebuie să vă luați după ceea ce fișierul tematic dinainte de 1989 încearcă să vă scoată ostentativ în față. Navigați prin fișier cu mult spirit critic. Până la urmă, sub o formă sau alta, fișele tuturor surselor sunt acolo. Încercați însă să înțelegeți unde ar fi putut fi plasate cele care vă interesează prin prisma obiectivelor proiectului dumneavoastră.

2.2.1.2 Rețeaua citărilor

Fișierul tematic realizat pe baza CZU stabilește legături între *teme*. Există însă și un alt mod de a detecta legătura dintre texte: trimiterele de la un text la altul.

E. Garfield este cel care, în 1955, a dezvoltat sistematic ideea rețelei de trimiteri de la un text la altul.⁷ Numărul de trimiteri la

⁷ A se vedea, în limba română, prezentarea făcută de către Mihailov, Ceornii și Ghilearevski[28, pp.165–175]. Numele lui Garfield apare însă abia la pagina 172. Autorii ruși au o atitudine rezervată față de metoda lui Garfield. Între altele ei cred că volumul de date ce trebuie prelucrate ar fi prea mare. Garfield a avut însă o intuiție corectă și capacitățile de prelucrare ale computerelor au rezolvat treptat problema.

2. La început a fost proiectul

un articol poate varia foarte mult. Poate fi menționat și de 500 de ori sau mai mult. Numărul mediu este de 1,5 menționări.⁸

**indicele
citărilor**

Institutul fondat de către Garfield publică faimosul *Science Citation Index*. În lumea academică, îndeosebi în SUA, indexul citărilor este un adevărat obiect de cult.⁹ Un autor are un rang înalt în ochii colegilor nu pe baza numărului de publicații, ci pe baza numărului de trimiteri ale altora la publicațiile sale.

Este inutil să spunem că nu se pune problema să construim manual, fără ajutorul unei armate de documentariști, un index al citărilor. Putem însă exploata principiul care stă la baza indexului citărilor pentru a ne construi bibliografia.

Sursele din bibliografia proiectului citează alte surse, pe care le includem în lista noastră bibliografică.¹⁰ În funcție de obiectivele proiectului nostru, verificăm dacă are sau nu rost să menținem în listă o sursă sau alta.

În faza următoare consultăm lucrările nou introduse în listă și identificăm citările din aceste lucrări. Din nou, facem o selecție în funcție de obiectivele noastre.

**diagrama
trimiteri-
lor**

Putem continua astfel până dăm de rădăcinile dezbaterii care ne interesează. Acestea s-ar putea să fie plasate uneori adânc în istoria filosofiei. Alteori, intervalul de timp va fi foarte scurt. Acesta nu înseamnă că și rețeaua de trimiteri este foarte simplă. Pentru o investigație serioasă se recomandă să alegem o rețea bogată de trimiteri.

O diagramă simplă (probabil pe o foaie A3) ne poate ajuta să dăm o formă mai inteligibilă rețelei de trimiteri.

Rețeaua de trimiteri, în afară de rolul ei în construirea bibliografiei, mai are un rost. Ea ne ajută să stabilim locul pe care o contribuție îl ocupă în literatura de specialitate.

Ce criteriu obiectiv avem pentru a aprecia dacă o lucrare este sau nu importantă? Opiniile subiective, faptul că unuia sau altuia – indiferent cine ar fi persoana respectivă – i se pare că lucrarea este importantă nu au o valoare prea mare. Trimiterile însă sunt ca un vot al membrilor comunității academice.

⁸ Conform estimărilor de la mijlocul secolului trecut.

⁹ V. situl consacrat lui Eugene Garfield la <<http://www.garfield.library.upenn.edu/>>.

¹⁰ Folosind această metodă putem construi mult mai mult decât o bibliografie. Eugene Garfield a arătat cum poate fi folosit indexul citărilor pentru a studia istoria științei. V. clasicul studiu al lui Eugene Garfield, Irving H. Sher și Richard J. Torpie, *The Use of Citation Data in Writing the History of Science* (Philadelphia: Institute for Scientific Information, 1964).

Nu vă bazați niciodată momentele-cheie ale eseului dumneavoastră pe o lucrări periferice într-o rețea a trimerilor.

În plus, rețelele de trimiteri vă pot da o idee obiectivă și despre punctele în care dezbaterile a fost deosebit de intensă.

2.2.1.3 Cercetările bibliografice

Cea mai simplă metodă de realizare a unei bibliografii o reprezintă desigur consultarea unui volum sau a unei publicații periodice în care găsiți o cercetare bibliografică pe tema care vă interesează.

Rezultatele unei cercetări bibliografice de specialitate înglobează, de obicei, sute de referiri. Dacă intrările din bibliografie au adnotări, atunci putem face mai ușor o selecție în funcție de obiectivele proiectului nostru.¹¹

2.3 Publicul cititor

Înainte de a trece la valorificarea roadelor muncii de documentare trebuie să vă puneți o problemă foarte importantă.¹²

Multe studente și mulți studenți scriu ca și cum ei ar fi un al doilea Wittgenstein. Înțelegerea textelor filosofului vienez cere o analiză atentă, de durată. Ca și aceste scrieri, eseurile studențești presupun uneori o reconstrucție migăloasă a gândurilor celor care le-au scris. Dacă muncești îndelung, s-ar putea să descoperi unele idei interesante și bine argumentate.

Eseurile de acest gen trădează lipsa de interes pentru o problemă fundamentală atunci când scrii: cui te adresezi? Cine te va citi?

Uneori este greu de răspuns la întrebarea de mai sus. Lucrurile nu stau însă așa în cazul eseurilor filosofice produse în vederea obținerii unui calificativ sau a unei note. Aici este limpede cine le va citi.

Prima observație este absolut evidentă: numărul de cititori este foarte mic. Lucrarea de seminar s-ar putea să fie în cea mai bună situație. Un număr de colegi, de colegi se vor uita peste ea sau vor audia susținerea ei în cadrul seminarului. Aceste colegi și acești colegi pot face observații extrem de utile. În orice caz, persoana care conduce seminarul va citi lucrarea. Dar aceasta nu înseamnă decât o singură cititoare sau un singur cititor sigur!

¹¹ Pentru istoria bibliografiei române puteți consulta Theodorescu[44].

¹² Paxson[34, pp.3–8] își începe ghidul cu un capitol despre publicul cititor. El subliniază rolul cunoașterii publicului și al alegerii limbajului potrivit cu publicul care va citi ceea ce scriem.

2. La început a fost proiectul

Lucrarea de diplomă va fi citită de cadrul didactic îndrumător. O va citi și referenta sau referentul anonim. O vor parcurge membrii comisiei. S-ar putea ca, în viitor, să mai fie consultată de către alte studenți sau studenți care lucrează la diplome sau de profesori care vor să ofere o sursă de informații pentru discipoli. Din nou, concluzia este foarte limpede: numărul cititorilor este extrem de mic.

Lucrarea prezentată la examen este cazul limită. Are doar un singur cititor sigur. Persoana care notează. S-ar putea ca ei să se alăture un alt examinator.

A doua observație este legată de contextul în care este parcurs eseu. Examinatoarele sau examinatorii au zeci, uneori sute de lucrări de corectat. Gândiți-vă c-ați avea în față un asemenea munte de lucrări! Iar timpul în care trebuie să terminați toată evaluarea eseurilor este foarte scurt.

În aceste condiții, trebuie să vă gândiți cât de importante devin elementele care ușurează lectura. Despre ele va fi vorba pe parcurs, dar este evident că o structură clară a lucrării, note, o bibliografie, indici sunt elemente care facilitează parcurgerea rapidă a eseului.

Un eseu-fluviu, filmul gândurilor care-i trec prin minte autorului, va fi foarte greu de urmărit. Cine evaluează eseu ar trebui să deslușească o structură, legături cu ideile altor autori și așa mai departe. În principiu, acest lucru nu este imposibil de făcut dacă ai foarte mult timp la dispoziție. Dar tocmai timpul este cel care presează.

Aceste observații conduc logic la o a treia chestiune, care-i interesează adesea pe studenți și studenți. Cât de mare să fie lucrarea?

**dimensiunile
eseului**

Există, din acest punct de vedere, o practică internațională. Lucrările de seminar sau cele pe care le prezinți la meditații au cel mult 2000 de cuvinte. Lucrările prezentate la examene au cel mult 3000 de cuvinte. O lucrare de diplomă ar trebui să aibă cel puțin 100 de pagini, iar una de doctorat cel puțin 200 de pagini.

Răspunsul veritabil la întrebarea cu privire la dimensiuni este totuși unul singur: *depinde!* Unele comisii de admitere sau de acordare a unor burse specifică dimensiunile eseurilor. Chiar dacă cerința este vagă, ceva de genul „dimensiune adecvată”, trebuie să gândim nu atât din perspectiva cantității, cât a structurării lucrării. O lucrare de dimensiuni mai mari, dar foarte bine structurată, cu trimiteri interne foarte judicioase gândite, cu indici se va citi mai ușor decât una mai scurtă, dar care nu este decât un text-fluviu, nesectionat, fără trimiteri interne, fără indici.

2.3.1 Ochiul critic al colegilor și colegelor

Publicul cititor al eseurilor studentești reprezintă o problemă și din alt punct de vedere. Există relativ puține reacții critice la ideile exprimate.

Ideal vorbind, seminariile ar trebui să ofere cadrul de dezbatere a ideilor din eseurile studentești. Timpul de seminar este însă și el, de multe ori, extrem de limitat. Nu toate proiectele și cu atât mai puțin eseurile într-o formă avansată de elaborare ajung să fie discutate în seminar.

În universități ca Oxford sau Cambridge există meditatori (*tutors*, în limba engleză) care au, între altele, misiunea de a discuta atent eseurile. Metoda aceasta este însă scumpă, din pricina personalului didactic numeros pe care-l presupune.

Puteți imita însă discuția cu meditatorul rugând o colegă sau un coleg să citească eseu dumneavoastră și să noteze observațiile pe care le are. Metoda funcționează într-un mediu în care studențele și studenții percep importanța unui ochi critic.

**imitarea
sistemului
meditațiilor**

Evident, serviciul de a scrie un referat cu observații critice trebuie să fie, direct sau indirect, unul reciproc.

Grație referatelor cu observații vă procurați ceva foarte prețios: un oponent sau o oponentă în carne și oase.

Metoda descrisă aici este bună, dar are și un risc. Discuția tinde să derapeze către „probleme profunde”. Ochiul mai puțin experimentat al colegilor sau colegelor s-ar putea să treacă prea ușor peste defecte ale documentării, lipsa notelor, bibliografia prost alcătuită și așa mai departe.

2.4 Construiți proiectul cu ajutorul computerului

Pentru a construi proiectul cu ajutorul computerului trebuie să studiați anexa B.1.3.1.1. Evaluați însă cu atenție diferențele dintre un proiect lucrat cu mâna și unul computerizat.

Un proiect computerizat permite trecerea graduală de la proiect la textul final. Practic, eu folosesc un fișier bib pentru primele elemente ale bibliografiei și comentarii¹³ pentru obiectivele proiectului. Comentariile sunt inserate în fișierul principal și, pe măsură ce proiectul avansează, în fișierele unde este plasat textul eseului.

¹³v. §B.1.3.2.3 pentru comentarii în fișiere electronice.

2. La început a fost proiectul

Folosesc, de asemenea, un simplu fișier text pentru titlul eseului și ideea-cheie. Acest fișier este de obicei creat primul, pentru a da un sens dosarului proiectului.

Partea II

Şantierul

Capitolul 3

Modularitatea manuscrisului

Cuprins

3.1	Secretul este rescrierea lucrării	46
3.1.1	Schițele care pregătesc tabloul final . . .	47
3.1.2	Rolul modulelor	47
3.1.3	Rescrierea Proiectului	48
3.2	Asamblarea modulelor	48
3.2.1	Structura arborescentă a textului	49
3.2.2	Parcurgerea arborelui textului	49
3.3	Modularizarea computerizată	50

Din practica proprie și din experiența acumulată pe parcursul îndrumării lucrărilor studentești, am constatat că nu există pericol mai mare decât acela de a începe scrierea eseului cu introducerea. În primul rând, nu prea știi cum va arăta lucrarea. În al doilea rând, ai o puternică tendință de a pune tot ce știi în introducere; parcă nu-ți mai rămân idei pentru restul lucrării.

Principiul înțelept este acela de a scrie mai întâi secțiunea-cheie a eseului. Dar pentru aceasta trebuie să existe o structură modulară a eseului. În acest capitol ne propunem să prezentăm modul în care este construită structura modulară și rolul ei.

Începeți prin a alcătui o listă cu ideile-cheie ale secțiunilor majore ale eseului. Dacă ar fi să aplic acest principiu în cazul cărții de față, atunci aș alcătui următoarea listă:

1. Vezi care este harta domeniului care te interesează și cum te poți orienta cu ajutorul ei;
2. Cercetează înainte de a scrie;

3. Scrie întâi modulul cel mai important;
4. Concluziile nu sunt nimic, argumentele sunt totul;
5. Arată-le și altora cum pot merge până la rădăcina ideilor;
6. Pune-te în pielea publicului cititor;
7. Textul va avea propria lui viață.

structura modulară Elementele acestei liste, cu unele comentarii suplimentare, au fost folosite la construirea, pentru partea de bază a acestei cărți, a unei structuri modulare.

Pentru fiecare modul există un număr de idei care sunt presupuse ca date. Ele nu sunt discutate în modulul respectiv. Dacă ideile presupuse de modul îți sunt familiare, atunci poți scrie sau citi modulul respectiv în mod independent.

3.1 Secretul este rescrierea lucrării

În ce constă însă investigația în cazul filosofiei? Este vorba oare doar despre umblatul prin biblioteci? Biblioteca este laboratorul persoanei care scrie un eseu filosofic, dar numai într-un sens restrâns.

Chiar dacă un eseu produs în cadrul procesului educațional este în primul rând dovada familiarizării cu o literatură, el nu este un colaj de fișe de lectură. Mai este nevoie de ceva.

Trebuie să analizăm critic propriile noastre soluții și argumente. Aceasta înseamnă că revedem periodic textul. Scriem o versiune a textului și apoi revizuim textul și producem o nouă versiune.

versiunile Versiunile diferite reflectă și progresul investigației. Cu alte cuvinte, de la un anumit punct, este greu de distins în mod absolut între cercetare și scris. Rescrierea lucrării este chiar procesul prin care încercăm să răspundem cât mai adecvat la problema pe care ne-am pus-o în eseu.¹

Putem compara elaborarea manuscrisului unui eseu cu pictarea unui tablou. Tabloul este realizat în straturi. Fiecare strat contribuie cumva la imaginea finală, dar foarte multe elemente din straturile de la început sunt acoperite de cele de deasupra. Versiunile manuscrisului seamană cu straturile. Fiecare versiune adaugă ceva și uneori elimină ceea ce era în versiunile anterioare.

¹De câte ori ar trebui rescris un eseu? Pryor[36, „at least 3 or 4”] este de părere că trebuie realizate 3-4 versiuni.

3.1.1 Schițele care pregătesc tabloul final

Când zugrăvește o scenă complexă, pictorul face schițe separate. Poate schița o mână, o figură umană, chipul unui câine și orice alt element al tabloului pe care vrea să-l studieze cât mai atent separat. Tot așa, persoana care scrie un eseu trebuie să recurgă la schițe. Eu folosesc fișe de lucru pe care le confecționez prin ruperea în două a unei coli de scris obișnuite.² Sau, mai simplu, îndoi o coală de scris în două. Pe aceste fișe de lucru dezvolt o idee. Încerc să-i prind liniile de forță și nu să dau contur unui text final.

Nu totul este cercetare în acest proces de elaborare a versiunilor succesive. Facem multe lucruri care nu au legătură directă cu cercetarea: finalizăm notele de subsol, indicii, revizuim din punct de vedere ortografic textul și așa mai departe.

3.1.2 Rolul modulelor

Rolul modularizării textului apare în adevărata sa lumină doar dacă suntem conștiente sau conștienți de importanța rescrierii textului. Ar fi aproape imposibil să rescii un text dacă n-ar avea o structură modulară. Voi folosi din nou exemplul cărții de față, dar vă puteți gândi la orice alt caz doriți. O carte care are în jur de trei sute de pagini nu poate fi rescrisă ca un tot. Nu poți tăia text dintr-o parte și lipi în altă parte fără să ai o structură modulară subiacentă.

Există o strânsă legătură între capacitățile memoriei de lucru a oamenilor și structurile modulare. Trei sute de pagini nestructurate ar depăși cu mult capacitățile memoriei noastre de lucru. Aceste ca-

**limitele
memoriei
de lucru**

pacități stau sub semnul numărului magic șapte, după cum a arătat George A. Miller, într-unul dintre studiile clasice ale psihologiei.³ Cartea de față are în total 11 capitole, ceea ce ar încălca regula numărului 7 (plus sau minus 2). Soluția pe care am găsit-o a fost aceea de a separa partea de bază de anexe. Ambele părți pot fi citite ca două cărți separate.

Dacă vă uitați la anexa B, vedeți că este foarte complexă. Ea este divizată însă într-o serie de module, pentru a face față complexității.

²Formatul folosit este A5. Vedeți aici §1.2.1.

³George A. Miller, „The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information“, *Psychological Review* 63:81–97 (1956) <<http://psychclassics.yorku.ca/Miller/>>.

3.1.3 Rescrierea Proiectului

Principiul elaborării mai întâi a părților substanțiale ale lucrării ne pune însă în fața unei noi întrebări. Dacă aceste părți sunt identificate în funcție de ideea-cheie a lucrării, cum găsim această această idee?

Aici intră din nou în scenă distincția dintre proiectul de cercetare și eseu. Am văzut că eseu trebuie rescris de un număr de ori. Proiectul de cercetare ar trebui și el refăcut pe parcurs? Răspunsul este imposibil de dat în termeni de da sau nu, pentru că întrebarea are două părți.

Da, proiectul de cercetare trebuie să sufere un proces de rescriere. n-are însă sens să refacem **pe parcurs** în mod radical proiectul. Dacă facem acest lucru, înseamnă că ne-am decis să scriem alt eseu. Putem însă ajusta din mers obiectivele proiectului.

Ceea ce trebuie cu adevărat să facem este o operație de optimizare a obiectivelor proiectului. Acest lucru trebuie făcut în faza inițială, când nu ne-am apucat să scriem textul propriu-zis.

Felul în care procedăm este relativ simplu. Rescriem obiectivele și subobiectivele proiectului în urma unei analize critice. Ceea ce căutăm să obținem este o legătură logică între obiective și un echilibru între ele.

Este probabil imposibil de indicat o regulă mecanică simplă pentru acest proces. Ceea ce am observat din practică este că, operând de un număr de ori rescrierea obiectivelor, apare o intuiție cu privire la ideea-cheie a întregului eseu.

culegeți	În cazul lucrărilor de diplomă este foarte important ca studenta
opinii	sau studentul să consulte cadrul didactic îndrumător în această
critice	fază a lucrului la proiectul lor. Conducătorul științific poate lucra
încă din	foarte greu cu o primă formă a textului. Observațiile sale pe text vor
faza	fi mai degrabă tangențiale. Nu vor viza elementele-cheie ale textului.
proiectu-	Motivul este foarte simplu: textul este amorf și lipsește o idee-cheie
lui	subiacentă.

3.2 Asamblarea modulelor

Modulele nu trebuie echivalate automat cu secțiunile textului. Abia când munca de redactare a eseului ajunge în faza finală conținutul modulelor este înghețat. Fiecare modul este transformat într-o secțiune a textului și primește un număr. Pasul următor este asamblarea modulelor.

Lucrând în mod tradițional unele module pot fi generate abia după faza în care există numerele secțiunilor și conținutul lor nu se mai schimbă. În această fază trebuie alcătuit cuprinsul. Când textul este paginat se trece la alcătuirea indicilor.

**părțile
standard
ale
textului**

Unele module au o poziție fixă. Trebuie ținut cont de faptul că textul este divizat în trei mari părți: partea frontală, textul principal și partea finală. Oricât de mic ar fi eseu, partea frontală trebuie să cuprindă o pagină de titlu și cuprinsul. Partea finală include bibliografia și indicii.

În cazul eseurilor studențești, pagina de titlu trebuie să cuprindă, pe lângă numele studentei sau studentului, anul și grupa din care fac parte. De asemenea, pagina de titlu include data la care a fost terminat eseu.

3.2.1 Structura arborescentă a textului

Când modulele au fost înghețate, secțiunile rezultate ar putea fi asemuate cu niște cutii pe care le punem una într-alta. O altă metaforă ar fi cea a arborelui.

Partea principală reprezintă trunchiul textului. În cazul unei cărți, din trunchi se desfac părți, iar părțile de despart în capitole. În cazul unora dintre cărți se trece direct la diviziunea pe capitole.

Eseurile pentru seminar sau examene, ca și articolele din reviste sau studiile din volume, nu se divid în capitole. Ele se împart în secțiuni și subsecțiuni.

Ceea ce deosebește o carte de un articol sau o broșură nu este dimensiunea ei mai mare, ci structura arborescentă mai complexă. Dimensiunea are o valoare pur orientativă. O lucrare de diplomă sau o disertație, care sunt cărți din punctul de vedere al structurii, ar trebui totuși să aibă minimum 80–100 de pagini.⁴

3.2.2 Parcurgerea arborelui textului

În partea principală, simpla însiruire a modulelor nu este suficientă. Structura eseului filosofic este neliniară. Trimiteri adecvate trebuie să arate de ce modul depinde modulul și care sunt drumurile de la un modul la altul.

⁴Mai multe elemente în legătură cu discuția, din perspectiva dimensiunilor, cu privire la distincția dintre broșuri și cărți se găsesc în monografia lui Mihailov, Ceornîi și Ghilearevski[28, pp.90–91].

În concluzie, trimerile sunt cheia asamblării modulelor. Textul nu este un simplu sac cu module numerotate.

trimerile Unii autori desenează chiar o diagramă a modului în care poate fi citită cartea lor. Trimerile ar trebuie să fie suficiente. Dacă m-am angajat pe o ramură a arborelui textului, o trimere bine gândită mă ajută să găsesc punctul la care mă pot întoarce în text. De acolo parcurg o altă ramură și urmăresc noile legături și așa mai departe.

Așezarea modulelor în arborele textului reflectă o structură. În fond, este structura „planului lucrării”. Atâta doar că ea poate fi conturată doar în etapa finală a lucrării. Până în etapa finală, ne folosim de **proiect** și de dezvoltările pe care le suferă treptat.

Când parcurgem textul este firesc să sesizăm **planul**, structura lucrării. Dacă structura lipsește, textul este o simplă colecție dezordonată de idei. Putem cădea însă lesne pradă iluziei că structura există integral de la bun început, nu că este realizată pe parcurs prin scrierea și rescrierea textului. Iluzia aceasta este deosebit de periculoasă pentru că poate crea un blocaj mental care le împiedică pe multe persoane să scrie.

3.3 Modularizarea computerizată

Marele avantaj al programului \LaTeX , prezentat în anexe, este acela că permite o trecere graduală de la modulele manuscrisului la secțiunile finale. De fapt, programul renumerează toate secțiunile din mers. Oricâte permutări ați face, programul este capabil să reasambleze totul pe parcurs.

Metoda descrisă pe scurt mai sus este imposibil de folosit în sistemul tradițional. Studiați mai ales în anexe secțiunea B.1.3.2.4 și evaluați atent avantajele computerizării.

Încercați să efectuați operațiile descrise în §B.1.3.2 cu ajutorul unui editor de birou. Comparați avantajele și dezavantajele perspectivei utilizării și cele ale perspectivei programării. Cred că perspectiva utilizării, cu accentul ei pe reglarea unor parametri, n-are în principiu cum depăși avantajele oferite de programare, unde puteți crea noi variabile și dirija procesele de prelucrare a datelor stocate în aceste variabile. Investiți timp și ceva bani în învățarea unui minimum de programare și veți recupera pe termen lung costurile resimțite poate dureros pe termen scurt.

Capitolul 4

Conținutul textului

Cuprins

4.1	Eseul ca încercare de a dezlega o problemă . . .	52
4.1.1	Deosebirea dintre o lucrare academică și propagandă	54
4.2	Limbajul filosofic	55
4.2.1	Terminologia filosofică	56
4.2.2	Propoziții și paragrafe	57
4.2.3	Utilizare și menționare	59
4.3	Argumentarea filosofică	61
4.3.1	Argumentele	61
4.3.2	Dizolvarea problemelor și distincțiilor .	62
4.3.3	Evitați ocolirea vicioasă a întrebărilor .	63
4.3.4	Evitați pseudoargumentul autorității . .	64
4.3.5	Explicații și exemple	64
4.4	Dezvoltarea conținutului	65
4.4.1	Argumentele pro și contra	65
4.4.2	Răspunsurile date oponentilor	65
4.5	Ajutorul limitat al computerului	66

Principiul de bază al acestui capitol și, de fapt, al întregii părți secunde a cărții este cel al separării conținutului de formă.

Separarea griii pentru conținut de preocuparea pentru formă este un principiu invocat adesea împotriva editoarelor de birou. Un asemenea editor lucrează după principiul tehnoredactării textului pe măsură ce este creat.¹ Forma frumoasă pe care o ia imediat textul în acest caz poate să ne dea iluzia că totul este în ordine în privința conținutului.²

Construirea unui text poate fi privită ca operația inversă rezumă-

¹Faimosul sistem WYSIWYG: ceea ce vezi pe ecran se va tipări și pe hârtie.

²Recomandarea noastră este să folosiți \LaTeX atunci când scrieți cu ajutorul

rii. De aceea este atât de important să învățăm să rezumăm textele altora. În acest fel descoperim cum poate fi creat un text.

Firește, din punct de vedere genetic, construirea textului este operația primară. Rezumarea lui de către alții este un proces ulterior. Dar logica plăsmuirii textului este mai ciudată.

Ceea ce am putea numi elaborarea **structurată** a eseului decurge strict după metoda inversării rezumării. Proiectul ne permite să degajăm ideea, firul conducător al eseului. Folosind acest fir vom identifica ideile-cheie ale părților, capitolelor și secțiunilor eseului. Coborâm apoi până la nivelul alineatelor și construim mai întâi ideile-cheie. Procesul ulterior constă în punerea pe acest schelet a substanței concrete a textului.

În practică, s-ar putea ca elaborarea textului să fie **orientată către componentele** sale. Există o idee a textului, dar aceasta nu este dezvoltată neapărat de sus în jos ca-n tehnica elaborării structurate a eseului. Fiecare componentă este dezvoltată separat, dar cu posibilitatea de a face legături cu restul textului. Pe măsură ce au fost create, componentele textului sunt asamblate. Avantajul elaborării orientate către componente a textului este acela de a ne putea concentra asupra componentelor centrale ale textului.

4.1 Eseul ca încercare de a dezlega o problemă

Multe eseuri sunt scrise ca și cum sarcina pe care și-au propus să o îndeplinească este aceea de a dezvălui un șir de adevăruri pe o temă dată.³ Eroarea comisă are cel puțin două aspecte. Aici nu vom discuta decât un prim aspect.⁴

Un eseu merită numele de „eseu” dacă respectă măcar ceea ce computerului. O introducere în \LaTeX , adaptată la cerințele unui eseu filosofic, găsiți aici în §B.1.3.

³Nu uitați că fiecare propoziție are o temă. Nu puține sunt eseurile studențești care conțin afirmații de genul „marele filosof X, personalitate cu un talent speculativ remarcabil și excepționale înclinații pentru ontologie, construiește un sistem impresionant prin arhitectonica sa și...”. Studenții imită în acest caz modele oferite de oameni mai în vârstă și sunt chiar convinși că au scris într-un stil elevat. Nu au făcut decât să înșiruie niște afirmații suspendate în aer. Normal este să explici de ce a creat X sistemul său, cum de a ajuns să fie preocupat de ontologie și care sunt legăturile dintre elementele sistemului său. Totul trebuie discutat pe fondul unor întrebări, pe care de multe ori și le-ar pune orice om care nu se lasă amețit de vorbe.

⁴Pentru cel de al doilea aspect al erorii a se vedea secțiunea consacrată rolului argumentelor în eseuri (4.3.1).

sugerează etimologia sa.⁵ Dacă înșiri sentențios niște enunțuri nu încerci, de fapt, să faci nimic. Deobicei repeți idei pe care le-ai găsit în cursul documentării.

Ca să încerci să faci ceva trebuie să-ți pui probleme, întrebări. Întregul eseu trebuie să fie o încercare de a răspunde la o problemă, dar – până la urmă – orice afirmație este din eseu este un răspuns la o întrebare. Cu alte cuvinte, presupuziția crucială atunci când scrieți chiar și cel mai lipsit de importanță alineat este aceea că încercați să clarificați o problemă.

**arta de a
pune
întrebări**

Este extrem de important să analizați problema pe care o supuneți atenției într-un punct sau altul al eseului, dar tot atât de important este să alegeți în mod judicios întrebările.

Eseurile filosofice, nu numai cele scrise de studenți sau studenți, suferă de o boală profesională larg răspândită: setea de a dezlega marile mistere. Cu cât întrebările sunt mai mari, cu atât este însă mai redusă șansa ca răspunsul să fie articulat și să poată fi examinat în mod riguros de către alții.

În orice caz, în eseurile pentru seminar sau în cele prezentate la examene, precum și-n teze sau disertații nu este pur și simplu loc pentru a ataca întrebări prea mari.

Firește este ca un eseu prezentat în seminar sau la examen să aibă întrebări subiacente care pot să pară mult prea simple, modeste. În ce a constatat disputa pe tema i , purtată între x , y și alții în perioada t , în contextul w ? Sau ce a vrut să spună z atunci când a făcut afirmația a ?

Chiar și-n lumea academică de limbă engleză, în ciuda numelui pompos de *research paper*, ceea ce se produce, de fapt, într-o lucrare studentască astfel intitulată este dovada capacității de a examina și de a prezenta în mod independent literatura pe o anumită temă.

Pentru filosofie, în orice caz, biblioteca este laboratorul. Chiar dacă, la început, ni se va părea ciudat, trebuie să începem cu cercetări în acest laborator. Pe măsură ce lucrăm în el s-ar putea să avem norocul să ne vină idei care nu le-au venit altora în probleme care au fost intens discutate.

Este foarte important ca problemele în care pretindem că avem o contribuție să fi fost intens discutate. Nu este foarte greu să inventăm probleme care nu au fost discutate anterior. Dar aceasta nu este decât o nișă pe care o construim pentru a pretinde c-am adus

**tratați
probleme
intens
discutate**

⁵În limba franceză actuală *essai* are sensul de „încercare”. Dacă mergem cu mai multe secole în urmă, cuvântul era folosit pentru cântărirea monezilor, deci punerea lor la încercare.

o contribuție originală. Stratagema nișei nu duce însă nicăieri. Este o simplă fundătură.

Boala profesională a întrebărilor mari este atât de răspândită încât cei din afara cercurilor filosofice nici nu cred că filosofi ar putea avea contribuții în probleme punctuale. De pildă, unii tind să vorbească despre „matematicianul”⁶ Jan Łukasiewicz care a inventat *scrierea poloneză*.⁷ Ideea scrierii poloneze este foarte simplă. În loc să notăm, să zicem, o adunare precum $2+2$ în mod tradițional, vom scrie același lucru, dar cu semnul pentru operație prefixat: $+ 2\ 2$.

Morala este simplă: nu căutați să răspundeți la întrebări mari; căutați să dați răspunsuri mari la întrebări punctuale.

Maxima aceasta este principiul călăuzitor al cercetării de mare anvergură. La nivelul lucrărilor care trebuie să primească o notă sau un calificativ ea este însă imposibil de aplicat. Evaluarea gradului în care un răspuns este „mare” sau nu ține de ceea ce se întâmplă în timp în dezbaterile de idei, precum și de implicațiile neașteptate ale soluției avansate. Scrierea poloneză, de pildă, nu este comodă când o utilizăm „manual”. Ea facilitează însă elaborarea de algoritmi pentru computere.

Ceea ce este important la nivelul unui eseu care primește o notă sau un calificativ este să existe o analiză corectă și independentă a problemei puse în discuție, precum și o înțelegere adecvată a debaterilor în problema dată.

4.1.1 Deosebirea dintre o lucrare academică și propagandă

Accentul pe probleme și discutarea argumentată a problemelor are o consecință extrem de importantă: eseurile academice nu sunt materiale de propagandă.

Reflexul propagandistului este să reducă totul la o reclamă în favoarea comunității sale. Uneori acest reflex ia forma reclamei pentru ordinea existentă într-o societate. Alteori propagandistul face reclamă pentru o lume ce există doar în viziunea adepților unei anumite credințe.

⁶De pildă, profesorii Michael Main și Walter Savitch vorbesc despre *matematicianul* Łukasiewicz în cartea lor *Data Structures and Other Objects* (Redwood: Benjamin/Cummings, 1995), p.318.

⁷În realitate, Łukasiewicz (1878–1956) a studiat filosofia la Lwów, Berlin și Louvain și toată viața s-a ocupat cu filosofia și logica. A scris, între altele, o carte celebră despre Aristotel. A se vedea în acest sens *Mała encyklopedia logiki* (Varșovia: Ossolineum, 1970), pp.162–163.

De ce sunt propaganda și filosofia atât de diferite? Există cel puțin două rațiuni ale acestei deosebiri.

Orice cercetare academică este neutră axiologic. Orice soluție este propusă în cadrul unei cercetări academice pentru a rezolva o problemă, nu pentru a fi în concordanță sau discordanță cu valorile sau credințele cuiva. Cercetarea academică nu urmărește nici să promoveze valori, nici să le submineze. Dacă eseul filosofic este unul de factură academică, atunci el trebuie să recurgă tot la argumente. Nu este o operă de promovare sau subminare a unor valori.

**neutrali-
tatea
axiologică**

Filosofia poate să ia foarte bine ceva ca dat. Ea își va pune însă întrebări cu privire la condițiile de posibilitate.⁸ Investigarea posibilităților este făcută fără a ține cont de restricții ideologice. Ea este efectiv o cercetare.

**analiza
condițiilor
de posibi-
litate**

Modul acesta de a proceda al filosofiei poate fi foarte derutant. Filosofia nu oferă răspunsuri în forma la care se așteaptă în mod curent oamenii. Nu este însă aici locul potrivit pentru a discuta acest lucru. Aceasta este treaba cursurilor de filosofie propriu-zise.

La urmă, dar nu în cele din urmă ca importanță, trebuie remarcat că orice modă intelectuală seamănă foarte bine cu un exercițiu propagandistic. Nu vă lăsați fermecate sau fermecați de modă.

Încălcarea cerințelor din această secțiune este lesne de detectat. Un eseu care este un lung șir de etichetări, de invective, de exclamații sau insinuări contrazice flagrant aceste cerințe.

4.2 Limbajul filosofic

Limbajul reprezintă o obsesie pentru filosofi de cele mai diverse orientări. S-au scris tomuri groase și nenumărate studii filosofice despre limbaj, inclusiv despre limbajul filosofic. Sub nici o formă nu ne propunem aici să abordăm probleme teoretice. Ne menținem la nivelul unor sfaturi practice elementare.

⁸ A se vedea celebrele întrebări kantiene privitoare la posibilitatea matematicii, fizicii etc. Ele nu vizează corectitudinea rezultatelor din matematică sau fizică, ci felul în care este posibil să le obținem. Pentru Kant, *Critica rațiunii pure* (București: Editura Științifică, 1969), p. 56, știința fizicii, de exemplu, este un dat. V. și I. Kant, *Prolegomene* (București: Editura Științifică și Enciclopedică, 1987), pp. 74-75.

4.2.1 Terminologia filosofică

Un eseu filosofic scris elegant n-ar trebui să fie plin de termeni tehnici. Cuvintele limbii obișnuite sunt cât se poate de utile și pentru filosofie. Ceea ce deosebește scrisul filosofic de utilizarea obișnuită a termenilor este însă caracterul critic al utilizării cuvintelor uzuale.

Să luăm drept exemplu un concept cum ar fi cel de libertate individuală. Încercați să vedeți cu un ochi critic care sunt consecințele unor idei obișnuite despre libertate. Dacă înțelegem prin libertate un mod de a-ți trăi viața fără a-ți bate capul cu calculele, fără planuri, atunci consecința directă este ideea că altcineva ar trebui să aibă grijă de bunăstarea individului, iar cea indirectă este caracterul benefic al unei dictaturi care ar avea grijă de indivizi. Astfel ar fi distrusă însă chiar libertatea indivizilor. Ceva nu este în ordine în ideea de la care am plecat.

Alternativa la ideea de libertate de mai sus este o strânsă relație cu ideea de responsabilitate. A face planuri și calcule intră în ideea de libertate și se corelează cu responsabilitatea individului. Consecința directă acum este cu totul alta decât cea de mai sus: important este ca alții să nu facă planuri în locul tău.

Termeni care în alte limbi sunt utilizați și-n vorbirea de toate zilele nu se lasă ușor împământenii însă în limba română. Un exemplu simplu este oferit de termenul englezesc *belief* a fost redat prin „opinie“, „judecată“, „crezare“, „crezământ“. Oricum l-ați reda, în cazul în care nu constatați existența unei redări constante în românește, scrieți o notă de subsol în care explicați alegerea făcută.

O altă problemă este cea a termenilor filosofici pe care nu-i înțelegem. Regula de bază este aici consultarea unui dicționar filosofic. Evitați pe cât puteți recursul la dicționarele cu caracter general.

Problema de mai sus este cu atât mai acută în cazul textelor în limbi străine. De multe ori trebuie să citești cărți întregi pentru a pricepe cum este utilizat un termen sau altul. Regula fermă rămâne însă cea a recursului cu prioritate la cărți, enciclopedii și dicționare de specialitate.

claritatea	O importanță la fel de mare, atunci când este vorba despre tra-
și	duceri, o are redarea constantă a termenilor tehnici. Traducerea lor
fidelitatea	în funcție de context sau folosirea unor sinonime (pentru ca să nu
au	se repete unele cuvinte!) este o rețetă sigură pentru dezastrele în
prioritate	plan logic. Un raționament perfect valid în original poate să sufere
	brusc, de pildă, de împătrirea termenilor din pricina maniei „scrisu-
	lui frumos“.

Morala este absolut clară: ceea ce contează nu este limbajul „ales“ și frumusețea stilului, ci claritatea și simplitatea exprimării.

4.2.2 Propoziții și paragrafe

Termenul „propoziție“ este folosit aici într-un sens considerabil mai larg decât în gramaticile de școală. Avem în vedere atât propozițiile simple, cât și cele compuse. Utilizarea aceasta corespunde cu cea din logică.

Mergând pe ideea că atunci când scriem refacem în sens invers drumul rezumării, putem spune că o propoziție este o pereche de forma „temă–comentariu“.

Comentariile trebuie să aducă o informație nouă, relevantă în contextul dat. O propoziție de genul „Kant a fost un filosof din secolul al XVIII-lea“ este (aproape) corectă, dar comentariul este neinformativ (din perspectiva obiectivelor unui eseu filosofic). Este informativ într-un manual, un dicționar sau o enciclopedie. Nu este informativ dacă discutăm, de exemplu, despre propozițiile analitice. Oricât ar părea de curios pentru mulți, propoziția respectivă are șanse minime de a fi corect construită în contextul eseului respectiv.

Propozițiile compuse sunt marcate de prezența unor elemente caracteristice. Din punct de vedere filosofic, sunt extrem de importante structuri cum ar fi cea de forma „dacă... , atunci...“ în care după „dacă“ se pune o condiție și după „atunci“ o condiție.

McCrimmon numește „balanță“ o structură în care o temă are două comentarii, corelate prin cuvântul „dar“.⁹

Dincolo de nivelul propozițiilor trebuie să examinăm modul de a construi paragrafe (alineate). Un mod foarte simplu de a construi un alineat este cel în care punem în paralel mai multe propoziții care au aceeași temă, dar comentarii diferite.¹⁰

O structură des întâlnită este cea în zigzag. Comentariul propoziției anterioare devine tema propoziției care urmează. Procedura este continuată în propoziția următoare și așa mai departe.¹¹

Distincția dintre propoziții compuse și alineate nu este foarte netă. Am putea, în fond, înlănțui propozițiile dintr-un alineat până le transformăm într-o singură propoziție compusă. Există vreo re-

⁹McCrimmon[24, p.147].

¹⁰Cf. ideea de propoziție paralelă așa cum o prezintă McCrimmon[24, pp.144–145].

¹¹Dubovik, Veize și Golovko[10, p.7] discută această structură la nivelul a două propoziții.

**folosiți
propoziții
concise** gulă pentru construirea propozițiilor noastre? Nu există o regulă cu caracter logic. Este recomandabil să scriem simplu, nu stufos. Este bine să folosim propozițiile economice. Propozițiile încâlcite dau o tentă de neclaritate argumentelor.

Scrieți propoziții cât mai scurte. Dacă treceți de 15 cuvinte într-o propoziție, încercați să vă opriți sau refaceți propoziția.¹² Folosiți structuri cât mai clare. Nu folosiți structuri excesiv de complexe.

**forma
activă este
prefera-
bilă** Propozițiile nu încep neapărat cu tema. Propoziția „profesorul a notat eseul” are tema la urmă. Unii lingviști opun aici tema agentului.¹³ În română, dacă trecem tema în poziție inițială trebuie să folosim apoi forma pasivă a verbului.

Mulți autori evită folosirea propozițiilor pasive. Strunk este de părere că propozițiile pasive nu se adresează direct publicului cititor și tind să fie lungi.¹⁴ Dacă scriem în limba engleză, recomandarea aceasta este aproape literă de lege. Ea se aplică și-n cazul limbii române dacă vrem ca propozițiile noastre să fie alerte și concise.¹⁵

**puneți
cheia la
începutul
alineatu-
lui** Care-ar fi structura optimă a unui paragraf? Judecați lucrurile prin prisma publicului cititor! Dacă este format din persoane care parcurg textul prin sondaj, este absolut recomandabil să plasați cheile paragrafelor la început.

Evitați paragrafele pivotante. Nu folosiți în nici un caz paragrafe cu cheia subînțeleasă.¹⁶

Strunk recomandă folosirea împarantezării în construcția paragrafului. Cheia este plasată atât la început, cât și la sfârșit. Strunk respinge în special alineatele care se termină cu o digresiune. Sfârși-

¹² „Capacitatea de a înțelege scade rapid după ce propoziția depășește 16 cuvinte” - apreciază psihologi de la Universitatea din Washington[37, p.1]. Dacă vrem ca eseul nostru să fie citit, trebuie să recurgem la propoziții scurte.

¹³ Exemplul cu eseul este adaptat după William O'Grady, Michael Dobrovolsky și Mark Aronoff, *Contemporary Linguistics* (New York: St.Martin's Press, 1993), p.194. Ei discută acolo, între altele, cazul interesant al limbii chineze. În chineză, simpla trecere a temei în poziție inițială și inserarea unei particule după ea transformă propoziția activă într-una pasivă.

¹⁴ Strunk[41, p.18]. Influența acestui punct de vedere este considerabilă. Corectorul gramatical integrat cunoscutului Microsoft Office sugerează întotdeauna trecerea de la forma pasivă la cea activă.

¹⁵ Diferența dintre „profesorul a notat eseul” și „eseul a fost notat de către profesor” nu este dramatică; propoziția pasivă are totuși 7 cuvinte, iar cea activă numai 4. În cazul unor propoziții mai complicate, forma activă este categoric preferabilă. Pasivul trădează și o nesiguranță în compoziția textului; s-ar părea c-am pus tema pe hârtie în ideea că vom vedea ulterior ce putem spune despre ea.

¹⁶ McCrimmon[24, p.109] formulează patru cerințe ale unui alineat: completitudine, unitate, ordine și coerență. Noi am optat pentru un sistem cu două criterii: existența unei singure idei-cheie; consistența logică.

tul vag, ceșos al alineatului ne împiedică să mai înțelegem cum este susținută ideea-cheie.¹⁷

Regula de mai sus se poate aplica oricărei secțiuni a unui text. În această carte, de pildă, urmărim legăturile dintre metodele tradiționale de elaborare a manuscrisului și tehnicile folosite pe computer. În consonanță cu această idee, încheiem secțiunea de față cu câteva observații privind utilizarea computerului.

Aș pleda în mod deosebit pentru scrisul pe computer în limbajul \LaTeX . În \TeX și \LaTeX fiecare propoziție poate fi scrisă pe un rând logic distinct.¹⁸ Paragrafele sunt separate de un rând alb. Sistemul acesta este ideal pentru organizarea logică a paragrafului. Scriem așa cum o facem în logică atunci când punem premisele unele sub altele. Putem marca foarte ușor (printr-un comentariu în \LaTeX , invizibil în textul final) cheia întregului paragraf.¹⁹

**compu-
terul și
lungimea
propoziți-
ilor**

Sistemul \LaTeX mai are un avantaj. Este ușor să apreciem lungimea propozițiilor. De exemplu, editorul cu care introduc textul afișează maximum 80 de semne pe un rând vizual. Dacă textul propoziției trece de un rând vizual și jumătate, am atins lungimea maximă a propoziției. Trebuie să trec pe următorul rând logic.²⁰

4.2.3 Utilizare și menționare

Să examinăm două propoziții:²¹

¹⁷Strunk[41, p.15]. A se vedea și exemplele care urmează până la pagina 18. La pagina 17, Strunk comentează un alineat din Lecky „The political value of history“. Alineatul respectiv are șase propoziții în care se susține că: în secolul al XVIII-lea a apărut o nouă concepție despre istorie (cheia alineatului); istoricii s-au orientat către explicarea fazelor proceselor istorice(explicarea cheii); condițiile care explică trecerea de la o stare la alta a procesului istoric au trecut în centrul atenției istoricilor; nu s-a mai scris istoria curților regale; au fost căutate lanțuri de cauze și efecte; istoricii au încercat să stabilească, pe baza experienței, care sunt condițiile de care depinde bunăstarea societății (consecință a cheii alineatului). Se vede aici foarte limpede structura unui paragraf construit prin *imparantizare*.

¹⁸Pentru distincția dintre rânduri logice și rânduri vizuale v. §A.2.1.2.

¹⁹Pentru detalii a se vedea §B.1.3.2.2.

²⁰Nu uitați că lungimea propozițiilor și a alineatelor este legată și de limitele memoriei noastre de lucru. În acest caz, este drept, trebuie să gândim în termeni de module, nu de cuvinte. Module sunt tema sau comentariul, de pildă. Sintagmele din care construim o temă sunt module. Propozițiile simple sunt module din care construim propoziții compuse. Un alineat compus din șapte module va solicita deja la maximum memoria noastră de lucru. V. aici nota 3 de la pagina pagina 47.

²¹Recomandările care urmează sunt preluate din Sparkes[40, pp.8–9]. Oricine este familiarizat cu logica știe că distincția dintre utilizare și menționare este una curentă în manualele de logică. S-ar putea să dați însă peste autori care se

1. Filosofia se deosebește atât de artă, cât și de știință.

2. Ar trebui să scriem „filosofia“ cu *z*, nu cu *s*.

În cazul primei propoziții s-ar putea să avem discuții cu privire la conținutul ei. Felul în care este **utilizat** cuvântul „filosofia“ nu pune însă probleme. În a doua propoziție nu utilizăm, ci **menționăm** cuvântul „filosofia“.

Folosirea ghilimelelor în a doua propoziție este absolut obligatorie. Gândiți-vă ce propoziție curioasă ar rezulta prin eliminarea ghilimelelor.

ghilimele Ghilimelele sunt unul dintre modurile de a evidenția faptul că menționăm ceva. O altă modalitate, folosită tot în a doua propoziție, este scrierea cursivă.

Ce statut au însă propozițiile de mai sus? Ele nu sunt utilizate, ci menționate. Faptul acesta este scos în evidență prin plasarea lor pe un rând distinct în pagină.

evidențierea înțelesurilor O problemă aparte o ridică înțelesurile. Acestea trebuie și ele evidențiate. Lucrul acesta se întâmplă atunci când, de pildă, explicăm înțelesul unui termen. Trebuie să scriem după modelul: termenul „validitate“ înseamnă, în cazul unei inferențe, că *ori de câte ori premisele sunt adevărate, concluzia este adevărată*.

În anexele cărții puteți vedea problemele speciale pe care le ridică menționarea unor termeni sau expresii din limbajele de programare. În cazul respectiv, textul este scris cu o garnitură specială de litere care imită caracterele mașinii de scris, folosită mai ales atunci când vrem să reproducem textul în facsimil.

aldinele În lumina acestei secțiuni ar trebui descurajată utilizarea cursivelor²² doar pentru a evidenția o idee care i se pare importantă autorului. Putem folosi pentru astfel de sublinieri literele **îngroșate** (aldinele)²³ sau sublinieri precum cele din dactilograme (o bară plasată sub text).

îndoiesc de distincția dintre utilizare și menționare. Recomandarea noastră este să nu dați curs acestor opinii.

²²Cursivele sunt utilizate când ne referim la înțeles sau când folosim un cuvânt dintr-o limbă străină în textul nostru.

²³Aldinele ridică o problemă de estetică a paginii. Parker[33, p.53] arată că aldinele folosite masiv creează impresia de tablă de șah. Cursivele, susține el, sunt asemenea cuvintelor rostite. Considerentele estetice nu primează totuși într-o lucrare academică. Folosirea cursivelor ar deveni derutantă dacă le-am atribui prea multe roluri.

Folosirea prea multor sublinieri retorice trebuie evitată. Abuzul de astfel de sublinieri face ca textul să devină mai degrabă derutant. Putem să spunem simplu și direct că un anumit lucru este important.

4.3 Argumentarea filosofică

Precauția cu care am abordat limbajul este valabilă și-n cazul argumentelor. Scopul nostru este doar să oferim câteva sfaturi practice elementare.

Argumentele joacă însă un rol deosebit în coagularea conținutului unui text filosofic. S-a spus uneori că filosofia este mai mult o artă de a pune întrebări. Acest lucru este în bună parte adevărat și a fost, indirect, susținut și de către noi.

Filosofia face însă ceva mai mult decât să pună întrebări. Ea testează modul în care se articulează argumentele în favoarea soluțiilor. Accentul este pus însă pe analiza argumentelor. Soluțiile în sine au mai puțină importanță. Pe scurt, argumentele sunt totul în filosofie, soluțiile nu reprezintă nimic.

Atitudinea descrisă mai sus contrastează vizibil cu cea din știință. Aici rezultatele, concluziile sunt importante. Ele sunt susținute cu grijă, sunt argumentate, dar analiza argumentelor ca atare și a presupunțiilor nu se află nicicum în centrul atenției.

Exemplul care ilustrează cel mai limpede afirmația de mai sus poate fi extras din matematică. Dacă matematicienii își concentrează toată atenția asupra unui mod de a argumente, cum ar fi raționamentul prin reducere la absurd, ajung să producă filosofie. După unii matematicieni, raționamentele prin absurd trebuie scoase din matematică. Alții susțin că ele nu prezintă un pericol deosebit. Din punct de vedere strict matematic, important este că există diferențe între rezultatele lor. Din punct de vedere strict filosofic, important este ce se spune despre modurile de argumentare ca atare.

4.3.1 Argumentele

Boala începătorilor în materie de eseuri filosofice constă în furnizarea de concluzii nesusținute de argumente. Concluziile sunt suspendate în aer. Eseul este o lungă serie de afirmații. Chiar dacă problema

**evitați
afirmațiile
neargumentate**

eseului este judicios aleasă, lipsa de argumente în favoarea soluției este un defect care distruge aproape întreaga construcție.²⁴

Nu este întâmplător faptul că filosofia și logica au fost strâns legate de-a lungul timpului. Chiar și un simplu alineat al unui eseu filosofic presupune familiarizarea cu logica. A miza pe un soi de simț înăscut al logicii textului nu este o soluție. Este doar un mod necultivat de a scrie un eseu filosofic.

Pentru a nu lăsa afirmația de mai sus în aer, am să încerc imediat să arăt cum cei care nu cultivă capacitățile de analiză logică nici nu știu ce anume pierd atunci când nu acordă suficientă atenție argumentelor.

Perspectivile asupra argumentării folosite în eseu filosofic pot fi de cel puțin două feluri. Dintr-o primă perspectivă, a argumenta înseamnă a face un efort pentru a produce o serie de premise care **sustțin** concluzia. O altă perspectivă este cea după care logica ar trebui folosită pentru a critica cât mai sever o afirmație. Dacă ea reușește să treacă testele criticii, atunci trebuie reținută.²⁵

Primul tip de argumentare ne angajează pe calea identificării unor fundamente. Filosofii care-l practică sunt, până la urmă, obligați să caute niște principii ultime.

Al doilea tip de argumentare focalizează eforturile filosofice în direcția criticii. Ca instrumente ale criticii filosofice sunt adesea folosite experimentele gândite.

4.3.2 Dizolvarea problemelor și distincțiilor

Există în istoria filosofiei secolului trecut o diferență celebră între Popper și Wittgenstein în atitudinea față de existența problemelor în filosofie. Popper sublinia importanța problemelor pentru filosofie. Wittgenstein respingea ideea că filosofia ar avea probleme: ea se ocupă doar cu **dizolvarea** problemelor.²⁶

²⁴Toate manualele de scriere a unui eseu filosofic din bibliografia de la sfârșit formulează cerința argumentării susținerilor avansate în lucrare. V., de exemplu, Pryor[36, „explicitly present reasons for the claims“].

²⁵Promotorul acestui stil de argumentare în filosofia secolului al XX-lea a fost Karl Popper. Din păcate, există prea puține cărți despre argumente care să explice acest mod de a argumenta. O excepție notabilă o constituie cartea lui Antony Flew[13, v. în special §§3.1, 3.15–3.20].

²⁶Între cei doi nu a avut loc niciodată o discuție veritabilă pe această temă. Popper, în *Unended Quest* (Londra: Routledge, 1992), pp.122-124, relatează incidentul celebru dintre el și Wittgenstein pe această temă. Când se contura un început de dispută între el și Wittgenstein, Popper a răspuns cu o glumă la solicitarea lui Wittgenstein de a-i da un exemplu de regulă morală. În acel

Disputa dintre Popper și Wittgenstein nu trebuie să ne deruteze. A dizolva o problemă este tot un mod de a obține un rezultat filosofic. Într-un fel sau altul, trebuia să-ți pui întrebări în legătură cu problema. Existența sau inexistența problemelor genuine în filosofie este o chestiune semnificativă din punctul de vedere al filosofiei ca atare. Ea nu schimbă însă modul în care trebuie să tratăm scrierea eseurilor filosofice.

Mai importantă este ideea de dizolvare a unei distincții filosofice. Orice încercare de analiză conceptuală implică fie trasarea, fie demolarea unei distincții. Lucrul acesta este firesc dacă ne gândim că a avea conceptul X antrenează capacitatea de a distinge X de ceea ce nu este X.

4.3.3 Evitați ocolirea vicioasă a întrebărilor

Întrebările sunt cruciale pentru un eseu academic. A le eluda în orice fel este o greșală. Uneori există doar iluzia că se răspunde la întrebare. Există o eroare²⁷ care se află la originea acestei iluzii.

McCrimmon oferă un exemplu simplu și clar: X vrea să împrumute bani de la Y pentru a crea o firmă; Y nu vrea să dea bani pentru că nu crede în profitabilitatea firmei; X argumentează că înapoierea banilor se va face de îndată ce firma obține profituri.²⁸

Eroarea pe care o comite X, în exemplul de mai sus, este aceea de considera drept adevărată tocmai susținerea care face obiectul discuției, al întrebării.

În eseurile filosofice este foarte ușor să comitem eroarea de a ocoli în mod vicios întrebarea. Să zicem c-aș vrea să dezvolt un argument împotriva ideii c-ar exista interese generale. Tentația cea mare este să arăt că interesele generale nu sunt decât interese individuale formulate cu pretenția de a reprezenta interesele tuturor. Cum demonstrez însă că lucrul acesta nu se întâmplă decât în unele cazuri? Nu pot susține că există doar interese individuale.²⁹ Aș ocoli în mod flagrant întrebarea, căci a susține că nu există decât interese individuale este același lucru cu a susține că nu există interese generale. Or, tocmai aceasta era problema în discuție.

Cum aș putea evita eroarea de mai sus? Opțiunea mea ar fi

moment Wittgenstein învârtea un vâtrai și Popper a dat drept exemplu de regulă „să nu ameninți oaspeții cu vâtraiul”. Wittgenstein s-a enervat și a părăsit sala.

²⁷ În limba engleză, se zice *begging the question*.

²⁸ McCrimmon[24, p.354].

²⁹ Dacă nu există decât interese individuale, atunci orice interese pretins general ar fi, de fapt, un interes individual.

pentru un mod de argumentare indirect. Nu atac chestiunea existenței intereselor generale ca atare, ci pe aceea a identificării lor. Ele trebuie identificate de către indivizi, iar indivizii se ghidează doar după interesele personale. Interesele celor puternici ar dicta atunci în procesul de identificare a intereselor generale. Este de presupus că oricine ar vrea să apeleze la ideea de interese generale ar susține și valoarea egalității dintre indivizi. Prin urmare indivizii ar trebui să aibă posibilitatea de a contribui în mod egal la determinarea intereselor generale. Dacă l-am încolțit astfel pe ipoteticul susținător al intereselor generale, pot trage concluzia că este superfluu să presupunem existența lor. Indivizii egali doar își pun în acord interese individuale.

4.3.4 Evitați pseudoargumentul autorității

O mare parte din literatura filosofică din perioada comunistă este minată de o eroare elementară. La tot pasul vezi exprimări de genul „așa cum au arătat clasicii marxismului” sau, mai rău, „după cum arată secretarul-general”.

eseul
trebuie să
probeze
capacita-
tea de a
gândi
independ-
dent

Persoanele care înșiruie presupuse adevăruri cred uneori c-ar câștiga ceva printr-o notă în care menționează o autoritate sau alta. Din păcate, nu s-au achitat astfel de obligația de a furniza un argument și de a-l discuta atent.

Chiar dacă argumentul pe care l-am găsi în scrierea citată ar fi absolut extraordinar, lucrurile tot prost stau. Eseul nu este o simplă înșiruire de fișe de lectură. Trebuie făcută dovada unei capacități de a gândi în mod independent, de a nu fi sclava sau sclavul argumentelor altora.

4.3.5 Explicații și exemple

S-ar putea să vă fi adus la disperare cu ideea de a nu repeta argumentele altora. S-ar putea să nu puteți efectiv găsi un argument mai bun sau unul diferit. Dacă vă aflați în această situație, atunci reproduceți argumentul găsit în literatură.³⁰ Adăugați însă propriile dumneavoastră explicații.

selectați
în mod
echilibrat
exemplele

Întotdeauna veți avea posibilitatea de a oferi explicații proprii. Însotțiți explicațiile date cu exemple și discutați exemplele date.

Atunci când dați exemple, încercați însă să meditați asupra nevoii de echilibru în selecția și analiza lor. Antony Flew discută contri-

³⁰Și menționați unde l-ați găsit!

buții la analiza filosofică a unor episoade ale războiului din Vietnam. El arată că, pe bună dreptate, masacrul de la My Lai³¹ este tratat ca un exemplu ce ridică numeroase probleme de răspundere morală. Pe de altă parte, el critică pe bună dreptate lipsa de echilibru a multor filosofi din mediul academic occidental al vremii. Aceștia refuzau să trateze în același mod serios pericolul reprezentat de regimurile comuniste.³²

4.4 Dezvoltarea conținutului

Un efect pervers al tendinței de a înșirui adevăruri pe o temă dată este incapacitatea de a trece dincolo de un număr foarte limitat de pagini. Studenta sau studentul care-și scriu lucrarea de diplomă vin disperați la consultație: au scris 20-30 de pagini și nu mai au nimic de spus!

Prezentăm în continuare două remedii la boala descrisă mai sus. Aplicate succesiv, ele ar trebui să permită dezvoltarea textului. Se spune uneori că, manevrate cu măiestrie, ele pot transforma un articol într-o carte de mari dimensiuni.

4.4.1 Argumentele pro și contra

Ideea de a cântări argumentele pro și contra unei soluții este mai mult decât un remediu pentru criza de idei. Este una dintre cerințele fundamentale ale unui eseu filosofic.

Fiecare argument trebuie expus distinct. Din §4.1.1 decurge limpede această cerință, precum și cerința de a nu evalua propagandistic argumentele. Nu evaluați argumentele în funcție de opțiunile dumneavoastră valorice sau, mai rău, în funcție de opțiunile valorice ale celor care vă evaluează eseu.

4.4.2 Răspunsurile date oponentilor

Este bine să nu scriem ca niște pustnici. Arătați eseu colegilor, consultați-vă cu cadrele didactice care vă îndrumă. Aceștia vor formula întrebări, obiecții. Răspundeți la aceste întrebări și obiecții și veți dezvolta eseu dumneavoastră.

Dacă nu dispuneți de oponenti în jurul dumneavoastră, căutați

³¹Sătenii din My Lai au fost uciși de către membri ai forțelor armate ale SUA.

³²Flew[12, p.55 și p.58]

obiecții formulate deja în literatura de specialitate la ceea ce vreți să susțineți. Încercați să arătați cum puteți depăși aceste obiecții.

Dacă nu găsiți în literatură unele obiecții, transformați-vă în propriul oponent.³³ Invocați în scris un posibil oponent și răspundeți.

Din punctul de vedere al structurii de ansamblu a textului răspunsurile la obiecții ridică o mare problemă. Oricine ar scrie cărți superbe cu această metodă, dacă n-ar trebui restructurat din mers totul.

Modulele textului trebuie revăzute pe măsură ce răspundem la obiecții. Tot timpul trebuie căutată o nouă arhitectură a ansamblului modulelor. Cu alte cuvinte, proiectul ca atare trebuie refăcut din mers. În principiu sună simplu, dar dacă este vorba despre o carte de 800 de pagini menținerea unei arhitecturi unitare a întregului argument ține de înalta performanță profesională.

4.5 Ajutorul limitat al computerului

Când este vorba despre conținutul eseului computerul vă poate fi de mult mai puțin ajutor. Practic nu este nimic de citit în anexe.

Nu disprețuiți însă automat programele de calculator. Nici cartea de față și nici o altă carte nu au cum să vă spună ce și cum să puneți în conținutul eseului.

Unele lucruri ar putea fi făcute cu ajutorul programelor de computer. Ați putea verifica, de pildă, validitatea inferențelor din text. Am putea folosi alte elemente ale inteligenței computaționale pentru a elabora conținutul textului. Aceste lucruri sunt însă cu mult peste nivelul la care sunt anexele.³⁴

³³Pentru tehnica anticipării obiecțiilor v. Horban[18, „anticipate objections“].

³⁴Detalii despre astfel de programe nu vor fi deocamdată pe situl acestei cărți.

Capitolul 5

Forma textului

Cuprins

5.1	Bibliografia	67
5.1.1	Forma listei bibliografice	69
5.1.2	Bibliografia ca un capitol de sine stătător	70
5.2	Notele	70
5.2.1	Citarea completă a surselor în note . . .	71
5.3	Trimiterile interne	76
5.4	Indexarea eseului	77
5.4.1	Tehnica tradițională a indexării	77
5.4.2	Indexarea cu ajutorul computerului	78
5.5	Lăsați L ^A T _E X să aibă grijă de formă	79

Ideea de bază a acestei cărți este să lăsăm computerul să aibă grijă de formă. Se pot spune foarte multe lucruri din această perspectivă; de aici și dimensiunea mai amplă a anexelor.

Persoanele care vor să aibă grijă de formă în mod tradițional ar trebui să consulte totuși anexele pentru a descoperi recomandările privitoare la stilul bibliografiei, al trimiterilor și așa mai departe.

Există însă o serie de elemente care trebuie învățate în forma lor tradițională, înainte de a ne apuca să utilizăm programe de calculator. La ele ne vom referi pe scurt în cele ce urmează.

5.1 Bibliografia

Un eseu academic se deosebește de alte scrieri mai ales prin forma sa. Sunt câteva lucruri care frapază pe oricine: bibliografia, notele, trimiterile.

Înainte de orice, pentru a alcătui bibliografia, avem nevoie de fișe bibliografice.¹

Din păcate, multe persoane cred că nu trebuie alcătuită o bibliografie. Ele confundă bibliografia cu o simplă listă a lucrărilor pe care le-au citat. Unii studenți îmi și spun câteodată că bibliografia eseului lor este implicit cuprinsă în notele de subsol.

Ce este eronat în acest mod de a proceda? Există o eroare de ordin practic. Ea face ca munca la eseu să fie complicată în mod inutil. Să zicem că persoana a citat în câteva locuri din eseu o carte și a folosit, de fiecare dată, o descriere bibliografică integrală. Între timp, descoperă o eroare în descrierea bibliografică sau vrea să facă o schimbare. Acest lucru este foarte greu de făcut.

Există, în mod tradițional, un fel de soluție la dificultatea de mai sus. Descrierea bibliografică integrală este dată o singură dată, iar în rest se folosesc metode de economisire a spațiului alocat notelor cu ajutorul vestitelor *op. cit.*, *loc. cit.*² Autorul poate corecta, în aceste condiții, ceva mai ușor. Prescurtările menționate fac însă textul dificil de parcurs de către publicul cititor. Prin urmare utilizarea lor masivă este total nerecomandabilă.³

Folosirea lui *ibidem* poate întinde și ea curse. Dacă intercalăm o notă când rescriem textul și ne referim la altă lucrare decât la cea menționată în notele pe care le separăm, atunci producem o trimitere eronată. Cred că *idem*, *ibidem* și *op. cit.* sunt lipsite de pericole doar dacă le folosim în cadrul aceleiași note.

Soluția cea mai elegantă este aceea de a avea, la sfârșit, o listă bibliografică și de a trimite la ea.

Există și un argument de principiu în favoarea distincției dintre lista bibliografică finală și bibliografia finală. Într-o carte precum cea de față, de pildă, are sens să folosești exemple dintr-o serie de documente tipărite de-a lungul timpului. Includerea lor în lista finală ar fi însă derutantă. Tema lor poate să fie cu totul alta decât tema principală a cărții. Menționarea lor, scoasă din context, ar conduce la ideea că ele sunt o sursă pentru ceea ce constituie elementele de bază ale cărții și nu doar un simplu izvor pentru trimiteri ocazionale.

Pe de altă parte sunt uneori situații în care o lucrare nu a fost menționată direct în eseu, dar a fost folosită intens pentru a înțelege

¹Citiți cu atenție secțiunea 1.2.2 pentru a afla cum trebuie elaborate fișele bibliografice.

²Pentru expresii din limba latină folosite în scrierile academice v. §6.1.6.

³Manualul Editurii Universității din Chicago[7, §15.41] descuraja categoric acest mod de a lucra chiar și-n 1969. Nu exista nici o legătură cu calculatorul în acel moment.

- Flew, Antony. 1975. "Issues in Teaching Contemporary Ethics." *Teaching Philosophy* 1 (1): 55-60 (Summer).
- Hairston, Maxine. 1974. *A Contemporary Rethoric*. Boston: Houghton Mifflin.
- Lamport, Leslie. 1994. *LT_EX: a document preparation system*. Reading, Massachusetts: Addison Wesley Longman, Inc.
- McCrimmon, James M. 1967. *Writing with a Purpose*. Boston: Houghton Mifflin.

Figura 5.1: Un fragment de bibliografie în stilul Chicago

dezbaterea în problema care face obiectul eseului. Este recomandabil să nu abuzăm de această posibilitate. O lucrare din lista bibliografică trebuie menționată cel puțin o dată în eseu. Aceasta este regula de bază.

Aceste premise ne permit să tragem concluzia că între conținutul bibliografic al textului propriu-zis și lista bibliografică finală nu există nicidecum o identitate. Lista finală permite publicului cititor să-și facă idee despre literatura de specialitate folosită pentru a elabora eseu.

5.1.1 Forma listei bibliografice

Există multe standarde în materie de liste bibliografice. Singura idee comună este până la urmă aceea de a prezenta în mod uniform blocurile de date din fișele bibliografice.

Editura Universității din Chicago folosește un sistem în care blocurile de informații sunt separate prin puncte. O versiune a acestui sistem o puteți studia în figura 5.1. Versiunea pură plasează anul în blocul de date referitoare la publicare, nu imediat după autor.⁴

**stilul
Chicago**

Avantajul listei în stil Chicago este că ne putem referi la o oricare dintre intrările ei folosind perechea formată din numele autorului și anul când a fost publicat documentul. De exemplu, putem scrie „Lamport(1994) oferă o trecere în revistă a sistemului L_AT_EX”.

Dacă lucrăm manual, stilul Chicago sau un stil analog sunt cele mai potrivite. Cred că este preferabil practicii (încă oficiale) din editurile românești. Standardul românesc nu separă blocurile de in-

⁴Pentru detalii trebuie consultat manualul Editurii Universității din Chicago menționat în bibliografia noastră[7, cap.16] sau o ediție mai recentă.

formații bibliografice. Toate datele unei intrări sunt despărțite prin virgule. Este evidențiat titlul, nu sursa.

Stilul Chicago are și el o serie de dezavantaje. Anul este uneori foarte derutant. S-ar putea să lucrăm cu o ediție recentă dintr-un autor de acum 300 de ani. Chiar și atunci când este sugestiv, anul încarcă textul cu o informație practic inutilă.

stilul simplu Este mult mai simplu să punem un număr de ordine în fața fiecărei intrări din bibliografie și să ne slujim de acel număr în trimiterile noastre.⁵ Recomandăm folosirea acestui stil simplu. Pentru a-l utiliza eficient aveți însă nevoie de un program de computer.⁶

5.1.2 Bibliografia ca un capitol de sine stătător

Autoarele sau autorii pot să urmărească și scopul de a transforma bibliografia cărții lor într-un capitol de prezentare a literaturii fundamentale pe tema dată. În acest caz, fiecare element inclus în bibliografie trebuie completat cu o adnotare.

adnotările Pentru a putea redacta adnotările este nevoie și de fișe de lectură. Este necesară cel puțin o fișă de lectură. Ideal ar fi să existe mai multe fișe de lectură. Pe baza lor, adnotarea va descrie pe scurt ideile-cheie ale documentului și relațiile dintre ele.

În acest caz, intrările bibliografiei pot fi grupate pe secțiuni. Fiecare secțiune va avea un titlu care evidențiază elementul comun tuturor documentelor din cadrul ei. Poate fi vorba, de pildă, despre ediții ale unui text fundamental, despre comentarii ale lui. Unele persoane preferă să separe documentele primare de cele secundare. Altele preferă să grupeze totul pe teme.

5.2 Notele

În tipografia tradițională notele de subsol erau destul de greu de realizat. Multe edituri pledau pentru note finale. Unele au păstrat acest obicei, deși astăzi nu mai are nici o justificare.

note de subsol Recomandarea noastră categorică este să utilizați note de sub-

⁵ Atenție însă la o posibilă eroare. Unele stiluri bibliografice folosesc abrevieri. Aceste abrevieri sunt folosite pentru a trimite la o intrare din bibliografie. n-are sens să puneți și numere, pe lângă abrevieri. Folosiți doar unul dintre sisteme: fie abrevierile, fie numerele.

⁶ Vedeți §B.2.3 și tot ceea ce presupune cunoscut secțiunea respectivă.

sol.⁷ Avantajul notelor de subsol este că sunt mult mai ușor de consultat decât notele finale.

Ce anume trebuie pus în notele de subsol? Există un principiu fundamental referitor la distincția dintre textul de bază și textul din notele de subsol. Textul de bază cuprinde idei formulate în termenii autoarei sau autorului. Este partea în care spui, ca autoare sau autor, ceea ce ai de spus. Notele de subsol sunt partea în care arăți ce au spus alții. Separarea aceasta trebuie să fie cât mai netă.

Recomandarea noastră ar fi să nu existe decât în mod excepțional citate în textul de bază. Citatele sau rezumatele ideilor altora trebuie să stea în notele de subsol. Aceste note sunt ca o prezentare documentară a literaturii de specialitate.

Notele de subsol mai au, de asemenea, funcția importantă de a face legătura dintre textul de bază și bibliografie. Din nou, recomandarea noastră este să nu îngropați în textul de bază trimiteri la bibliografie. Locul lor este în subsolul paginii, acolo unde stau firesc ideile altora sau trimiterile la ideile altora. Folosim îngroparea în text doar atunci când acest lucru contribuie la creșterea fluentei frazei. În acest caz este vorba despre exprimări de genul „cartea lui X[număr] este...“.

Cât de ample pot fi notele de subsol? După părerea noastră, ele pot fi chiar foarte ample. Ele reprezintă, în acest caz, o prezentare documentară detaliată a dezbaterilor de specialitate. Mari autori, precum Karl R. Popper, un om cu idei originale absolut remarcabile, cultivau notele de subsol ample.

5.2.1 Citarea completă a surselor în note

Atunci când o sursă bibliografică nu este inclusă în bibliografia finală ea trebuie citată complet în notele de subsol.

Ideal ar fi însă ca o asemenea sursă să fie menționată o singură dată. Menționarea ei repetată este un semn c-ar trebui să o includem în lista bibliografică.

Stilul menționării surselor bibliografice în notele de subsol diferă de cel al bibliografiei. Prezentăm în continuare opțiunile noastre.⁸

⁷Notele acestea sunt extrem de ușor de creat cu ajutorul programelor de computer. Studiați aici §B.1.3.4.

⁸Opțiunile noastre pentru cărți, studii și articole reprezintă o adaptare a soluțiilor din Manualul Chicago[7, §15.34 ș.u.]. Am păstrat chiar și virgula de după titlul articolului. Plasarea datei este însă diferită.

5.2.1.1 Citarea unei cărți sau a unui studiu dintr-un volum

Formatul recomandat de noi pentru citarea unei cărți în notele de subsol sau în notele finale este următorul:

Prenumele Numele (autoarei sau autorului etc.),
Titlul cărții (Localitatea: Editura, anul), p. sau pp.

Titlul cărții este evidențiat pentru că el coincide cu numele sursei documentului.

datele privind publicarea Sugestia noastră este ca faptele referitoare la publicare să fie puse într-o paranteză. Stilul parantezei este cel al Chicago University Press. Motivul pentru care recomandăm să se procedeze așa este legat de menținerea, pe cât posibil, a fluenței textului. Dacă referirea este îngropată într-o frază, paranteza rotundă și stilul ei specific ne arată imediat că este vorba despre fapte legate de publicarea cărții.

două puncte după localitate Recomandăm separarea datelor referitoare la localitate de rest prin două puncte. Care este avantajul punctuației care recurge la cele două puncte? Este foarte greu uneori să deosebești între localitate și editură. De exemplu, în bibliografia finală am folosit inițial doar virgule. După un timp, când m-am uitat la descrierea cărții celor doi Lester[22], am fost și eu derutat. Apărea acolo secvența „Glenview, Scott, Foresman and Company“. Am adăugat statul american unde se află localitatea Glenview, dar lucrurile nu păreau să devină mai clare. Cele două puncte rezolvă problema.

Când punem statul în care se află localitatea? Când localitatea este mai puțin cunoscută sau când există o ambiguitate. Exemplul clasic este „Cambridge“. La ce se referă acest nume? La localitatea din Anglia sau la cea din Massachusetts? Trebuie precizat.⁹ Evident, n-are sens să pui „Anglia“, după „Londra“, când cartea a fost editată de Routledge.

Dacă vreți să folosiți stilul tradițional al editurilor românești, despărțiți toate elementele prin virgule. Consecvența este însă o regulă absolută. Folosiți același stil peste tot.

Dacă trebuie menționat ceva cu privire la versiunile textului sau traducere, puneți aceste informații (separând totul prin virgule) după titlul cărții.

culegeri de studii Dacă este vorba despre un volum care are un editor, acest lucru trebuie evidențiat prin (ed.) sau (editori). Autorii vor fi mențio-

⁹Manualul Chicago University Press[7, §15.90]

nați doar atunci când trimitem la studiile lor. Formatul trimiterii, în acest caz, este:

Prenumele Numele (autoarei/autorului), „Titlul studiului“ în Prenumele Numele (editoarei/editorului) (ed.), *Titlul volumului* (Localitatea: Editura, anul), p. sau pp.

Dacă lucrarea este inclusă în lista bibliografică finală, punem doar numele persoanei care a editat cartea, urmat de o paranteză dreptunghiulară cu numărul de ordine din lista bibliografică, separat printr-o virgulă de paginile la care trimitem.

Titlul studiului este pus în ghilimele. Titlul volumului este subliniat. Volumul în care este inclus studiul este **sursa** bibliografică. Descrierea sursei este precedată de *in*.¹⁰

Menționarea paginilor poate fi înlocuită cu un element structural (pagini standard pentru textele clasice, numărul secțiunii sau chiar al alineatului). Aceste numere trebuie să existe însă efectiv în sursa citată.¹¹ Altfel, publicul cititor poate fi derutat sau numerotările executate manual ar putea să nu coincidă. De asemenea, numerele trebuie să trimită la porțiuni suficient de mici. Trimiterile la un capitol întreg sunt justificate doar atunci când avem în vedere firul conducător pentru capitolul respectiv.

Cred că nu este o eroare să punem un „la“ sau chiar „la pagina“ și să închidem între virgule referirea la pagină sau elementul structural, dacă toată referirea este îngropată într-o propoziție. Fluența textului ar avea de câștigat, iar numărul de semne suplimentare este minim.

5.2.1.2 Citarea unui articol

Dacă nu vreți să includeți un articol în bibliografie, dar îl menționați în text recomandăm următorul stil:

Prenumele Numele (autoarei, autorului etc.), „Titlul articolului“, *Denumirea revistei* volumul(numărul): prima pagină–ultima pagină (luna anul) <Adresa pe Internet, dacă este cazul>

¹⁰Folosirea lui *in* pare preferabilă atunci când bibliografia ar putea fi consultată și de către persoane care nu cunosc limba română. Acest cuvânt are sens în latină și este larg folosit în limbile europene. Pentru evoluția (mai complicată decât credem de obicei) de la latinescul *in* la „în“ v. Iancu și Horvat[19, p.143].

¹¹De exemplu, în cartea lui Antony Flew[13] fiecare alineat este numerotat.

5. Forma textului

Stilul recomandat este compact și permite menținerea fluentei textului, în cazul în care referirea este îngropată într-o propoziție.

Unele reviste nu furnizează toate elementele de mai sus. Nu trebuie să lipsească în nici un caz anul sau paginile. Dacă aveți doar informații cu privire la volum, puneți doar volumul. La fel și-n cazul numărului.

două puncte Cele două puncte sunt importante. Ele permit o exprimare compactă și atrag, în plus, atenția asupra faptului că este vorba despre o revistă, nu despre o carte. La cărți, cele două puncte sunt puse ~~după~~ ^{după} numele editurii.

evidențiată sursa Pledăm categoric pentru practica internațională care cere să fie evidențiată sursa. Sistemul standard românesc (cel în care titlul articolului este subliniat și denumirea revistei este pusă în ghilimele) este eronat. Revista este sursa documentului la care facem trimitere.

Evident, din nou, regula absolută este cea a uniformității stilului. Chiar dacă **ați** evidențiat titlul articolului și **ați** pune în ghilimele denumirea revistei sau **ați** folosi o altă variantă, faceți acest lucru în mod consecvent.¹²

5.2.1.3 Citarea unei surse de pe Internet

Recomandarea noastră, în cazul documentelor de diverse tipuri disponibile pe Internet, este să se folosească următorul stil în trimiterile complete inserate în note:

Prenumele Numele (autoarei, autorului etc.), *Denumirea sursei* [notă despre document] <Adresa pe Internet>, data (data accesării: ziua/luna/anul), cheie.

data Data versiunii documentului o scriem în forma: ziua, luna (în litere), ~~anul~~ ^{anul}¹³. Ea este o dată oferită de către autoare sau autor! Nu puneți nici o asemenea dată, dacă nu găsiți una pe situl Internet citat.

data accesării Data accesării este, strict vorbind, facultativă. Este însă deosebit de utilă, dacă există riscul ca situl respectiv să dispară între timp sau să-și modifice conținutul. Forma datei accesării este diferită de cea a datei versiunii și urmează modelul 15/11/2003 (observați folosirea exclusiv a cifrelor, precum și barele oblice).

¹²Pentru un exemplu de trimitere completă la un articol v. nota 1 de la pagina 183.

¹³Un exemplu ar fi, de pildă, data de 14 octombrie 2003.

Nota cu privire la textul documentului n-am folosit-o o bună vreme. Sunt însă două rațiuni importante care impun utilizarea ei. Pe de o parte, documentele de pe Internet au un caracter foarte diferit. Este util uneori să explicităm acest caracter. De asemenea, are sens în unele cazuri să precizăm situl Internet unde se găsesc. Pe de altă parte, nota este introdusă din rațiuni estetice. Adresele de pe Internet sunt uneori foarte lungi. Nu pot fi despărțite în silabe. Lasă din această pricină spații albe absolut inestetice. Nota are o lungime reglabilă și permite eliminarea acestor spații.

**nota
despre
document**

În nici un caz nu trebuie să lipsească adresa de pe Internet. Punerea ei între paranteze ascuțite ni se pare necesară. Altfel se pot ivi confuzii. Este punctul final o parte a adresei? Este cifra care urmează o parte a adresei? Pentru a evita astfel de nedumeriri trebuie folosite paranteze ascuțite. Ele marchează limpede începutul și sfârșitul adresei.

**adresa pe
Internet**

Unele documente de pe Internet nu au titlu. De aceea ne-am referit la denumirea sursei. Așa ceva trebuie să existe sub o formă sau alta. Preferabil este un element de tipul titlului tradițional al unui text. Orice am folosi, trebuie să fie un element pe care să-l putem găsi și cu ajutorul unui motor de căutare pe Internet.

titlul

Ce punem la rubrica intitulată „cheie”? Dacă nu este vorba de o carte electronică, documentele de pe Internet nu au număr de pagină. Unele reviste electronice sau biblioteci electronice notează numărul paragrafului. Folosiți acel număr, dacă el există.

**locul la
care
trimitem
în sursă**

Dacă paragrafele nu sunt numerotate, dar sunt numerotate secțiunile textului, atunci putem trimite la numărul secțiunii.

Adesea lipsesc însă orice fel de numere pentru secțiuni sau paragrafe. Dacă fișierele HTML sunt rezonabil de mici putem trimite la ele.¹⁴ Dacă nu, pentru a trimite la un anumit punct din text, trebuie folosit un mic fragment din text (pus între ghilimele). Acest fragment trebuie redat absolut exact, pentru a putea fi folosit într-o căutare electronică. Trebuie verificată însă lipsa sa de ambiguitate. Trebuie să nu existe două sau mai multe puncte din text în care apare fragmentul respectiv.¹⁵

¹⁴Deși se numesc „pagini”, ca și „paginile” cărților, documentele HTML se compun de obicei din numeroase fișiere conectate prin trimiteri.

¹⁵Pentru un exemplu de trimitere completă la un document de pe Internet v. nota 17 de la pagina 97.

5.3 Trimiterile interne

Evidențierea legăturilor dintre elementele textului este o parte importantă a artei de a scrie un eseu lizibil. Pe de o parte, n-are rost să repetăm ceea ce am spus. Pe de altă parte, are sens să atragem atenția asupra faptului că o idee continuă să fie discutată într-un moment ulterior.

Eseurile academice nu sunt romane. Ele nu se citesc capitol cu capitol, fără a sări ceva. Nu este vorba despre frivolitate aici. Motivul este mult mai profund și rezidă în geometria structurii textului.

Doar un eseu scurt și relativ simplu are o structură lineară. În acest caz, secțiunile eseului nu pot fi parcurse decât în succesiunea lor lineară.

structura arborescentă Eseurile mai complexe au însă o structură arborescentă. Există un trunchi comun, dar din acesta se desprind o serie de ramuri. Dacă avem un anumit interes cognitiv atunci când citim textul, are sens să mergem doar pe un anumit drum prin textul respectiv.

Din perspectiva celor care scriu textul, drumurile posibile trebuie trasate la nivelul proiectului. Când elaborează efectiv componentele textului vor trebui însă să se gândească la felul în care ele se vor combina în așa fel încât drumurile prin text să nu se înfunde.

Există mai multe tipuri de trimiteri. Un prim tip de trimitere este cel către o altă pagină din text.

Un alt tip de trimitere, acolo unde este nevoie de mai multă precizie, este către pagină și rândul din pagină în care începe trimiterea. Acest gen de trimiteri este exemplificat și-n anexele cărții de față.

Al treilea tip de trimiteri sunt cele către o secțiune a textului. Pentru a realiza efectiv trimiterile de acest tip trebuie să numerotați secțiunile.

Alte trimiteri se fac la formulele, tabelele și figurile incluse în text.¹⁶

Deși tehnic sunt posibile, trimiterile de la o notă la alta cred c-ar trebui folosite doar atunci când nu vrem să repetăm ceva comentat pe larg. Notele sunt asemenea unor paranteze în textul de bază. Saltul dintr-o paranteză într-alta poate să fie extrem de derutant pentru publicul cititor.

Pentru toate aceste tipuri de trimiteri se recomandă folosirea lui *vide*. De obicei folosim forma prescurtată a lui *vide* (*v.*).

¹⁶În anexe găsiți informații despre toate aceste tipuri de trimiteri.

5.4 Indexarea eseului

Indexarea este o profesie. Cu un oarecare efort, autoarele sau autorii pot să creeze totuși un indice acceptabil.

Este un indice absolut necesar? La o lucrare amplă, de factură academică, da! Doar editurile românești au curiosul obicei de a face economii la acest capitol.

Indicii sunt mult mai utili decât cuprinsul când trebuie găsit în text un anumit concept. Sunt, de asemenea, de mare folos când vrem să ne croim propriul drum prin carte.

5.4.1 Tehnica tradițională a indexării

Care este marele dezavantaj al autoarei sau autorului atunci când indexează? Cunoașterea prea bună a textului! Și, de multe ori, atașamentul prea mare intens față de text. Le este greu să aibă o lectură panoramică.

În principiu, trebuie folosite tehnicile descrise în cazul rezumării. Creați un mic proiect pentru indice. Faceți-vă o idee cu privire la cuvintele-cheie și ierarhia lor. Nu fiți prea zeloase sau prea zeloși. Indicii trebuie să reflecte liniile de forță ale textului.¹⁷

Nu trebuie să confundăm indicele cu o statistică a cuvintelor folosite în text. Pentru aceasta se folosesc programe speciale. Există posibilitatea de a detecta și contextul în care sunt folosiți anumiți termeni.¹⁸ Toate aceste lucruri le fac însă cei care studiază un text.

Ghidul lui Vickers[46] distinge o serie de etape ale indexării. Practic, este vorba despre: identificarea descriptorilor, trimiterile la pagini, sortarea intrărilor din indice, redactarea indicelui.

Recomandarea mea este să folosiți fișe de format mic (A8 sau A7) pentru a realiza indicii. Pe fișe treceți descriptorul și pagina la care apare. Folosiți un creion sau un evidențiator pentru a marca pe pagină locul indexat. Scrieți pe margine descriptorul (nu cuvintele din text!).

Vickers distinge între trei categorii de descriptori: numele de persoane, numele de locuri, conceptele.¹⁹ Recomandarea noastră

indice vs. concordanță

fișe pentru indici

descriptori

¹⁷Mi s-a întâmplat să pierd multă vreme pentru a găsi pe o pagină discuția în fugă a unei idei. Ea fusese în mod inutil menționată în indice. Chiar numele proprii trebuie puse în indice, cred eu, doar atunci când se spune ceva mai substanțial despre persoana respectivă.

¹⁸Eu unul prefer termenul de „concordanță” pentru asemenea operații cu textul. A se vedea anexa A.4.1.

¹⁹Vickers[46, descri.htm].

este să folosiți tehnicile descrise în secțiunea 1.3 pentru a detecta conceptele-cheie.

Vickers atrage, pe bună dreptate, atenția asupra pericolului ca un anumit descriptor să fie urmat de un număr foarte mare de pagini.²⁰ Eu aș recomanda să nu fie decât cel mult șapte trimeri la pagini.²¹

**subdes-
criptori** Dacă avem prea multe trimeri la un descriptor, soluția este să-l transformăm într-o cupolă mare (un modul mai cuprinzător) și să distingem mai mulți subdescriptori (modulele care sunt puse sub cupola modului mai mare).

De exemplu, dacă descriptorul „Kant, Immanuel” trimite la prea multe pagini, distingem subdescriptori, cum ar fi „judecăți analitice la Kant”. Putem omite „la Kant”, dacă nu se produc confuzii. Unii pun o linioară sau o tildă pentru descriptorul principal. Nu cred că sunt foarte utile.

sortarea După ce ați terminat de alcătuit fișele pentru indice, treceți la sortarea lor. Nu folosiți în nici un caz o listă. Va fi foarte greu de sortat.

**trimeri
în
interiorul
indicii** Pentru a dirija atenția către alt descriptor se folosesc intrări în lista de indici de forma „Aquino, *vezi* Toma din Aquino”²². Descriptorul la care se face trimitere trebuie să trimită la pagini, în nici un caz nu la alt descriptor. Nu aș recomanda folosirea masivă a acestei metode și nici sistemul cu „*vezi și*”. Textul ca atare va sugera publicului cititor ce intrare din indice să mai consulte.

Lista articolelor din indice trebuie creată abia în faza finală. În mod normal, lista este scrisă pe două coloane. Verificați prin sondaj acuratețea indexării. Evident, dacă totul este în ordine, nu se pune problema de a mai clinti din loc vreo pagină din text. În condițiile indexării manuale, tot indicele s-ar da peste cap.

5.4.2 Indexarea cu ajutorul computerului

Mulți editori se fereșc de indexarea tradițională. Indicii trebuie alcătuiți când manuscrisul este gata. Acest lucru întârzie apariția cărții și sporește costurile.

Cred că în spatele argumentelor obiective, de genul celui de mai sus, există și o presupuziție subiectivă. Există la unii oameni cre-

²⁰Vickers[46, subdes.htm].

²¹Vickers[46, subdes.htm] recomandă șase trimeri.

²²V. exemplele din Vickers[46, refpas.htm], cu adaptările necesare la practica din textele românești.

dința puternică în lectura liniară a textului. După ei, textul trebuie citit cu sfințenie, de la început până la sfârșit.

Presupoziția este falsă. Lectura textelor academice este prin excelență neliniară. De multe ori, o carte, cum este și aceasta, este de folosit mai ales ca lucrare de referință. Căutăm, din când în când, o soluție în ea. Comparăm soluția noastră cu cea din carte, dar nu studiem totul de la început până la sfârșit.²³

Computerul ne poate ajuta să depășim și dificultățile inerente indexării. Indexarea ca atare continuă să fie o operație predominant manuală. Există însă unele avantaje substanțiale oferite de către computer.

**avantajele
indexării
compute-
rizate**

În \LaTeX indicii pot fi elaborați și din mers. Nu este neapărat nevoie ca paginile să fie „înghețate“. Citiți anexa B.5 pentru a afla cum se indexează în \LaTeX .

Dacă-ți recurs la comentarii invizibile pentru a dezvolta proiectul, atunci aveți deja conceptele-cheie pentru alineate. Trebuie doar să le alegeți pe cele mai importante și să le grupați cu ajutorul proiectului de indice.

5.5 Lăsați \LaTeX să aibă grijă de formă

Sfatul nostru este categoric: folosiți sistemul \LaTeX ! Chiar dacă veți investi considerabil mai mult timp decât în cazul unui editor de birou, studiați anexa B.1.3.

Din practica îndrumării lucrărilor studentești pot spune că o mare problemă o reprezintă gestiunea notelor de subsol în cazul editoarelor de birou. Cât timp notele nu sunt numeroase, lucrurile decurg normal. Am văzut însă lucrări la care totul era dat peste cap de către procesorul de cuvinte.²⁴

O altă problemă o ridică diferențele dintre rularea pe calculatoare diferite sau chiar pe același calculator a fișierelor editoarelor de birou. Rezultatele diferă!

\TeX , marele program al lui Donald Knuth, este capabil să calculeze nu „la milimetru“, ci la fracțiuni extrem de mici de milime-

²³Indici se alcătuiesc și pentru documentația în format electronic. A se vedea în acest sens Manualul Sun[43, pp.149–178]. Indicii nu pot fi supliți de simpla căutare prin textul electronic. Indicii sunt o imagine a cheilor textului.

²⁴Credem că multe dintre deficiențele publicațiilor realizate pe computer relevate pe bună dreptate de Funeriu[14, pp.9–17], pot fi mai lesne îndreptate folosind \LaTeX .

tru. Gestiunea notelor este impecabilă. Efectele se văd din plin și-n L^AT_EX.

semnifi- Dacă vă decideți să lucrați manual, trebuie să folosiți indicațiile
cația de pregătire a dactilogramei din manuale scrise înainte de pătrun-
dactilo- derea computerelor în lumea tiparului și a editurilor.²⁵ De exemplu,
gramei când faceți trimiteri trebuie să lăsați spații albe pentru numărul
secțiunii, al paginii etc. și să le completați abia când paginile sunt
bătute în cuie.

²⁵Capitolul al 2-lea din manualul editurii Universității din Chicago[7, pp.25–64] este extrem de instructiv din acest punct de vedere.

Partea III

Arta supraviețuirii

Capitolul 6

Cu ochii redacției

Oricât ar suna de pompos, textul are propria sa viață. Uneori este citit cu mult după moartea autoarei sau autorului. Enorma majoritate a textelor pe care le avem în vedere în această carte nu au o soartă chiar atât de glorioasă. Vor fi totuși citite în mod sigur de alții. Trebuie să supraviețuiască unor priviri de multe ori grăbite, dar versate.

**supra-
viețuirea
la o
lectură
grăbită**

Cuprins

6.1	Stilul eseului filosofic	83
6.1.1	Titlurile	84
6.1.2	Ideile altora	86
6.1.3	Evitarea limbajului imprecis	94
6.1.4	Nu excludem pe nimeni	95
6.1.5	Semnele de punctuație	96
6.1.6	Expresiile latinești	101
6.1.7	Termenii limbii grecești vechi	104
6.1.8	Cuvintele înșelătoare	104
6.1.9	Exprimarea distincțiilor filosofice	105
6.2	Corectura	108
6.2.1	Literele	109
6.2.2	Ortogramele	109
6.2.3	Semantica textului	110
6.2.4	Gramatica	111
6.3	Computerul și faza finală a elaborării textului	112

6.1 Stilul eseului filosofic

Există un principiu fundamental pentru un eseu academic, de orice factură ar fi el. Cred că numele lui cel mai potrivit este „principiul sobrietății“. Un text academic este simplu și funcțional.

stilul simplu Mulți nu pricep importanța simplității. Am primit odată un mesaj prin poșta electronică de la cineva care susținea că pagina mea de web nu este atractivă. Persoana respectivă se oferea să-mi facă o pagină care arată ca lumea. Nu mai țin exact minte ce am răspuns. Am cedat totuși presiunii psihologice și m-am apucat să construiesc o altă pagină. Din fericire, cred eu, am schimbat doar poarta de intrare în pagina de web.

Am rămas cu poarta creată atunci. Eu unul m-am obișnuit cu ea, dar cred, în continuare, că nu trebuie făcut rabat de la simplitate când este vorba despre eseul academic ca atare.

Cele scrise mai sus se aplică, *mutatis mutandis*, eseurilor filosofice. Principiul sobrietății este valabil atât pentru stilul în care sunt scrise, cât și pentru stilul în care sunt tehnoredactate.

Într-un eseu filosofic detaliile trebuie cizelate din perspectiva logicii argumentelor. Calofilia nu este deloc dezirabilă.

Un exemplu extrem de limpede îl oferă cuvintele care se repetă. Evitarea repetițiilor poate să înlăture un neajuns minor, dar să distrugă un argument. Premise care aveau termeni comuni vor fi brusc deconectate. Va trebui să ne întrebăm care sunt relațiile dintre sinonimele introduse. Tot felul de discuții pseudosavante se pot ivi pe tema unor presupuse nuanțe produse doar în încercarea de a obține cu orice preț un text frumos.

empatia Înainte de orice verificați cum sună textul dumneavoastră pentru alții. Încercați să vă puneți în locul lor! Înțelegeți ce este scris în text? Reușiți să regăsiți informațiile la care se face trimitere în text? Vă simțiți înstrăinată sau înstrăinat din pricina unora dintre formulări? Aveți sentimentul că n-ar trebui să faceți parte din publicul cititor? Și, nu în ultimul rând, sunt prezente în text greșeli de ortografie supărătoare?

6.1.1 Titlurile

Fiecare secțiune a textului trebuie să aibă un titlu. Titlul trebuie să acopere cât mai precis tema secțiunii respective. Din acest motiv titlurile academice sunt destul de lungi.

titluri precise Este foarte important ca titlul să sugereze exact discuția avută în vedere. Un titlul precum „Școala austriacă și problema dovezilor empirice” este potrivit dacă se discută despre atitudinea membrilor Școlii austriece față de dovezile empirice. Este cât se poate de nepotrivit să folosim drept titlu, într-o asemenea situație, „Știința economică și dovezile empirice”. Chiar dacă suntem absolut convinse sau

convinși că punctul de vedere al Școlii austriece este valabil pentru știința economică în genere, nu trebuie să dăm un astfel de titlu.

Un titlu precum „Știința economică și dovezile empirice” ne-ar obliga să prezentăm ce au zis toți economiștii despre dovezile empirice și, mai rău, toți filosofi. Mai mult decât atât, nu există de fapt nici o discuție care să fie acoperită de către formula din titlu.

În altă ordine de idei trebuie spus că titlurile care conțin metafore sunt seducătoare, dar trebuie recurs la ele cu reținere. Cred că pot fi folosite doar atunci când vrei să extinzi aria publicului cititor.

**metaforele
și riscurile
lor**

Un alt motiv pentru folosirea metaforelor îl reprezintă dorința de a atrage atenția asupra ideilor-cheie. Când scrieți o lucrare pentru o comisie gândiți-vă însă și la riscurile metaforelor. S-ar putea ca titlul să fie interpretat cu totul altfel decât doriți. Recurgeți mai degrabă la titluri academice simple și precise.

6.1.1.1 Titlurile altora

Titlurile cărților, articolelor sau ale altor documente la care ne referim în text trebuie redată ca atare. Nu operăm modificări. Dacă nu folosim titlul, atunci utilizăm o expresie de genul „în cartea sa despre...”

Titlurile cărților, articolelor sau altor documente în limbi străine se redau în original. Putem adăuga o paranteză în care traducem titlul. Recomandarea mea ar fi să utilizați cu mare prudență această posibilitate. Pe de o parte, traducerea titlurilor nu este chiar așa de ușoară cum pare. Pe de altă parte, titlurile nu sunt ele ca atare informații, ci sunt precum niște indicatoare către locul în care găsim informații. Acestea sunt însă utile doar pentru cine știe limba textului la care ne referim.

Atunci când este vorba despre o traducere avem posibilitatea să punem în paranteză titlul originalului. Tehnica aceasta poate fi utilă mai ales în bibliografii sau note de subsol. A se vedea aici exemplul intrării numărul [23]. În cazul cărții lui Evan Mawdsley și Thomas Munck, s-ar putea ca multă lume să nu știe maghiară, dar să poată consulta ediția în limba engleză.

6.1.1.2 Denumirile firmelor sau produselor

Denumirile firmelor sau produselor (inclusiv ale programelor de calculator) trebuie tratate ca și titlurile. Nu se traduc. Nu uitați că firmele au drepturi de proprietate asupra acestor denumiri.

În textele cu caracter comercial se folosesc semnele speciale ©, ® sau TM. De pildă, titlul ghidului Windows98 cuprinde secvența *Microsoft® Windows® 98*. În ghidul Sun pentru editarea documentelor electronice întâlnim expresia *FrameMaker® templates for the book*¹

Semnele ® etc. se pun la prima utilizare a unei denumiri. ² În textele academice aceste semne sunt rar folosite. Nu uitați însă că, în aceste texte, respectul pentru titlurile, textele și ideile altora este literă de lege.

Conform practicii din textele academice, putem folosi și prescurtări ale unor denumiri deja menționate.

6.1.2 Ideile altora

De nimic nu trebuie avut mai multă grijă într-o lucrare academică decât de ideile altora. Aceste idei sunt proprietatea lor privată și ea trebuie respectată.

N-are sens să deschidem aici o discuție despre proprietatea intelectuală. Chestiunea este privită aici din unghiul de vedere al practicii.

6.1.2.1 Ideile care plutesc în aer

Dacă am stabili regula că orice idee care nu-ți aparține trebuie însoțită de o notă de subsol în care menționezi de unde ai preluat-o, această regulă ar fi de neaplicat în practică. Lucrul acesta se întâmplă cel mai des când încerci să pui pe hârtie standardele nescrise. Nu puține exemple sunt și-n cartea de față.

Cum citim când vrem să corectăm un text? Ideea că trebuie să evităm capcanele lecturii care urmărește să sesizeze înțelesurile (dar pierde literele) am auzit-o de alții. Evident, cu foarte mulți ani în urmă. Chiar dacă mi-aș aminti unde am auzit-o, lucrul acesta n-are nici o importanță. Nu-mi scriu autobiografia în notele de subsol. Oricum, cei de la care am auzit ideea au auzit-o și ei de la alții. Ar fi inutil, în acest caz, să mergem până la rădăcina lucrurilor. Nu facem istoria corecturii.

**citați doar
pe baza
unui text**

În plus, cred că nici nu este corect să-i citezi pe alții în situa-

¹ *Read Me First!*[43, p.xiii].

² Detaliile sunt discutate în ghiduri speciale precum *Read Me First!*[43, pp.61-64].

ția când n-ai un text scris în față. Există riscul de a le atribui o formulare pe care poate n-ar accepta-o.³

Ce facem în asemenea cazuri? Cred că lucrul cel mai important este să nu emitem pretenția c-am făcut vreo descoperire. Totul este să nu ne atribuim cine știe ce merite când este vorba despre un lucru cunoscut de multă vreme.

Faptul că nu suntem în contact cu un text scris ne ferește însă de un mare pericol. Sub o formă sau alta, trebuie să exprimăm ideea așa cum ne pricepem.

Regula aceasta trebuie aplicată însă cu mare grijă. Când avem îndoieli trebuie să verificăm și să facem trimitere de rigoare. Am să dau un exemplu chiar din cartea de față.

Voiam să menționez deviza de pe frontispiciul Academiei din Atena. Mi-am amintit formula care circulă curent: să nu intre ni-meni care nu este geometru. Formula nu se potrivea tocmai bine cu ceea ce voiam eu să sugerez. Ceva nu este de altfel în regulă în logica internă a traducerii: doar matematicienii făceau parte din școala lui Platon? Am dat cu destulă greutate de o formulare în limba greacă veche, însoțită de traducerea convențională. Cuvântul-cheie era însă acolo ἀγεωμέτρητος. De aici totul a fost mai ușor. Regula este să te duci la un dicționar explicativ „grec-grec“. În dicționar sensul era *necunoscător într-ale geometriei*. El se potrivea cu traduceri din dicționarele bilingve. Mai rămânea să găsesc un text vechi în greacă în care să apară expresia și o traducere în românește. Rezultatul îl puteți vedea la pagina 24.⁴

**orice
informație
trebuie
verificată**

Există și situația când știm la ce facem aluzie, dar n-are sens să facem o trimitere. Am să dau un exemplu tot din cartea de față. „La început a fost proiectul“ cuprinde o aluzie transparentă la Sfânta Scriptură. N-are noimă să faci note de subsol în asemenea situații. Ar fi absolut ridicol să explici aluzia în notele de subsol. Tot efectul pe care voi fi sperat să-l obțin folosind titluri metaforice s-ar topi. Distrugerea metaforei poate conduce chiar la afirmații pe care nici nu vrei să le faci. Voiam doar să sugerez existența unei anumite ordini a creației și doar atât.

³Situația aceasta trebuie deosebită însă limpede de cazul în care este vorba de o discuție foarte precisă. Cineva poate să vă comunice o idee despre o teorie. Ideea respectivă nu plutește în aer, ci are un autor sau o autoare. Este absolut incorect să ți-o atribui. Trebuie spus de la cine ai ideea, cum ți-a fost transmisă și când.

⁴Lucrurile s-au petrecut chiar așa. Desigur, aș fi putut să-mi amintesc direct de traducerea din David. Chiar dacă aș fi avut o asemenea străfulgerare, metoda normală este cea descrisă mai sus.

În acest ultim caz, mizăm pe faptul că oricine știe la ce se face aluzie. Este chiar important ca ideea să fie lăsată să plutească în aer.

6.1.2.2 Parafrizarea

Când preluăm ideile altora nu folosim neapărat cuvintele lor. Parafrizăm formulările. Trebuie să facem însă o trimitere. Ideea nu ne aparține și știm unde să trimitem publicul cititor, dacă vrea să afle mai multe pe tema respectivă.

**împarantezarea
parafrazărilor** Unde plasăm însă trimiterea? James D. Lester, Sr. și James D. Lester, Jr. observă, pe bună dreptate, că trebuie să existe o împarantezare: începem cu o exprimare de forma „autorul X susține că...”; încheiem cu o trimitere.⁵

Dacă parafrizarea este pierdută prin alineat, atunci nu este limpede unde începe ideea altcuiva și unde se termină.

Împarantezarea descrisă mai sus este un bun sistem și atunci când facem rezumatul unei porțiuni ample dintr-un alt text sau chiar al unei lucrări întregi. Simplele trimiteri la sfârșitul unui alineat sau al unei secțiuni nu sunt suficiente.

6.1.2.3 Extrasele din textele altora

O regulă orientativă ar fi aceea de a cita surse primare. Evident, porțiunea extrasă trebuie pusă în ghilimele și trebuie să se încheie cu o trimitere la sursă, inclusiv la pagina (sau paginile) de unde a fost extras citatul.

În mod normal sursele secundare trebuie parafrazate. Anumite surse secundare, cum sunt enciclopediile, trebuie menționate, dar n-are sens să dăm citate.

apud Dacă dispuneți de citat prin intermediul altei surse, atunci trebuie să deosebim cu atenție textul de sursă. Sursa trebuie precedată de cuvântul *apud*.⁶

⁵Lester, Sr. și Lester, Jr.[22, pp.89–90].

⁶Am căutat în zadar prin bibliotecă exemple de utilizare semnificativă a lui *apud*. *Apud* era metoda omului amărât din perioada dinainte de 1989, când accesul la surse era drastic limitat. Există însă situații când n-avem decât fragmente sau redări indirecte ale ideilor. Este cazul multor autori antici. N-avem cum să-l cităm pe Pericle. Trebuie să ne folosim de Tucidide. Este oare o sursă bună? Popper crede că Tucidide a fost de față când Pericle a ținut faimosul său discurs (v. Karl Popper, *Societatea deschisă și dușmanii ei* [București: Humanitas, 1993], nota 16 de la cap.6). Oricum ar sta lucrurile, n-avem o sursă directă.

Ideea *apud*-ului nu este complicată, dar îi pune pe mulți în încurcătură pentru că nu fac deosebirea dintre text (document) și sursă. Dacă ai găsit un extras dintr-un autor în cartea sau articolul altcuiva este ca și cum ai găsi un articol întreg într-un volum colectiv.⁷ Folosiți trimiterea care însoțește extrasul pentru a menționa datele ca-n cazul articolului. Particularitatea rezidă acum în punerea lui *apud*. Restul este descrierea normală a sursei în care ai găsit extrasul, cu precizarea inclusiv a paginii la care se află extrasul.

Cum redăm citatele din sursele primare? Înainte de 1989, editurile din România aplicau următoarea regulă: trebuia recurs la traducerea românească. Mulți dintre colegii mei critică, pe bună dreptate, această regulă. Ea reflecta mentalitatea conform căreia există neapărat o singură traducere și aceasta trebuie să fie traducerea oficială.

**contra
traducerilor
oficiale**

Cred că trebuie să distingem, atunci când dăm citate din sursele primare, între pasajele simplu de tradus și cele dificile. Are sens să redăm în limba în care au fost scrise doar pasajele dificile, a căror traducere este problematică.

Trebuie să oferim, atunci când cităm în altă limbă, și o traducere românească? Cred că da! Traducerea echivalează, în fond, cu o interpretare a textului. Dacă nu suntem pregătite sau pregătiți să oferim o traducere, atunci trebuie în orice caz să explicăm modul în care înțelegem noi textul.

**traducerea
citatelor**

Netraduse ar trebui să rămână doar acele scurte pasaje pe care le oferim pentru a permite căutarea în textele electronice. Traducerea mai mult ar încurca lucrurile decât ar folosi la ceva în cazul respectiv.

Trebuie știut că legislația care protejează drepturile de autor interzice operarea, sub orice formă, de extrase abuziv de lungi. La aceasta se adaugă normele tacite ale comunității academice. Nu poți să pui o carte întreagă între ghilimele și să faci cu seninătate o trimitere. Lucrul acesta pare de sine înțeles.

Cât de lungi pot fi extrasele citate în propria lucrare? Răspunsul aproximativ ar fi „câteva rânduri“. În SUA, unde editurile sunt foarte stricte în această privință trebuie cerută permisiunea expresă a editurii chiar pentru citate relativ scurte. Normele diferă însă de la editură la editură.

**lungimea
citatelor**

Chiar dacă este vorba despre un text din domeniul public sau despre un text pus la dispoziție spre citare pentru oricine, tot nu trebuie să exagerăm. Trebuie să ne punem întrebarea „de ce cităm pasajul respectiv“? Cel mai frecvent îl cităm pentru a susține o ipo-

⁷ V. aici §1.2.2.2 pentru cazul studiilor din volume colective sau din reviste.

teză, pentru a arăta unde am detectat o anumită nuanță în concepția pe care o analizăm. Ne oprim acolo unde simțim c-am acoperit cu dovezi susținerile noastre.

Cred că plasarea sistematică în note a extraselor de orice fel subliniază clar rolul lor de dovezi în sprijinul ideilor afirmate în textul de bază. Extrasele nu fac parte din structura de rezistență a textului.

para- În timp părerile mele cu privire la dimensiunile și numărul citate-
frazarea lor au evoluat. Acum cred că parafrazările sunt preferabile citatelor,
este iar citatele scurte sunt preferabile celor lungi.

prefera- În lucrările studentești lungimea citatelor ar trebui să fie extrem
bilă de limitată. Examinatorii trebuie să evalueze calitatea însușirii unor
idei și a prelucrării lor. Citatele trebuie date numai atunci când
credeți c-ați descoperit ceva absolut deosebit. În rest sunt suficiente
trimiterile neprecedate de extrase.

6.1.2.4 Tranziția la respectul pentru regulile academice

Orice încălcare a regulilor din secțiunile 6.1.2.2 și 6.1.2.3 constituie un plagiat.

plagiatul Plagiatul nu este comis doar atunci când un text este preluat ca atare sau este tradus și inclus fără ghilimele în textul altuia. Preluarea ideilor, fără trimiterile de rigoare, este la fel de gravă.

Includerea unui extras nemarcat prin ghilimele, dar încheiat cu cu o trimitere la sursă, este un mod neonest de a cita. Publicul este indus în eroare și crede că are de a face cu o simplă parafrază sau preluare liberă a unei idei.

Preluarea ideilor este relativ greu de documentat. Preluarea unui extras, fără trimiterea de rigoare sau fără ghilimelele convenite, este mult mai lesne de documentat. În ciuda opiniei răspândite în mod curent după care era textelor electronice ar facilita plagiatul, detectarea plagiatului în texte electronice este mult mai facilă. Ea poate fi făcută în mod automat. În măsura în care ambele texte sunt disponibile pe Internet, lucrul acesta îl poate face oricine știe să utilizeze un motor de căutare.

Mai gravă mi se pare plagierea structurii unui text. Acest lucru este mult mai greu de detectat. Cuvintele ar putea fi complet diferite, dar ideile și modul în care sunt conexe în document sunt similare.

Ce este așa de rău în plagiat? Se pot da multe răspunsuri la această întrebare. Putem examina chestiunea din punct de vedere

juridic, din punct de vedere etic, dar aici nu ne-am propus așa ceva. În ciuda aparențelor, răspunsurile la aceste întrebări nici n-ar fi teribil de relevante în contextul de față.

Nu vreau să pedalez pe apelul, larg răspândit prin regulamentele universităților, la onestitatea academică. Există un alt aspect important al întregii chestiuni: trăsăturile caracteristice comunității academice. Întrebarea bună privește caracteristicile distinctive ale comunității academice.

Mi-e greu să decid dacă trimiterile la scrierile altora este trăsătura care distinge în mod categoric comunitatea academică⁸ de alte comunități. Neutralitatea academică, de pildă, este și ea un lucru crucial. Cert este că trimiterile la scrierile altora sar în ochi atunci când te uiți la produsele comunității academice.

În afara comunității academice, trimiterile nu sunt o practică răspândită. Chiar revistele de popularizare a științei, care sunt o interfață între comunitatea academică și lumea externă, conțin mai puține trimiteri decât o publicație de specialitate. Articolele de ziar nu conțin decât în mod excepțional trimiteri. Romanele cu trimiteri ar fi ceva ridicol. Cum ai putea face trimiteri într-o poezie.⁹

Dacă medităm ceva mai atent, ne dăm seama de importanța trimiterilor. Plagiatul în sensul de preluare integrală sau parțială a unui text este respins și-n lumea artei. Din punctul acesta de vedere, există un cerc mai larg al comunităților care nu acceptă plagiatul în sensul tare al termenului. Dar nici lucrul acesta nu este universal valabil.

Există comunități în care ceea ce ar fi plagiat în comunitatea academică sau în lumea artei este o un mod de a proceda acceptabil. Politicienii nu-și scriu discursurile. În orice caz, cei mai ocupați nici nu și-ar putea scrie singuri toate discursurile. Numele membrilor echipei lor sunt menționate doar discret și arareori știm cine ce discurs a scris. Vedetele nu-și scriu autobiografiile. Persoanele care le-au ajutat să-și transforme amintirile într-un text fluent sunt menționate, dar nu sunt figurează, de regulă, drept coautori.¹⁰

Adevărul este că nu e nimic neînregulă în practicile descrise mai

⁸ „Comunitatea academică“ se referă aici nu doar la lumea universităților, ci la lumea cercetătorilor și universităților.

⁹ Ar fi foarte curios să le cerem poetilor să facă note de subsol în care să precizeze de unde au preluat tema, ce i-a inspirat, cine a mai avut aceleași sentimente. Ar fi tot atât de curios să le ceri așa ceva cum ar fi curios să le ceri să dovedească în chip științific că au avut o trăire sau alta.

¹⁰ În comunitatea academică, în asemenea cazuri, toată lumea trebuie să apară printre autori.

**trimiteri
la scrierile
altora**

sus. Textele au fost comandate așa cum comanzi un costum de gală și sunt menționate persoanele care poartă costumele respective.

comunitatea academică Recursul la trimiteri, identificarea atentă a surselor extraselor și ideilor este o trăsătură distinctivă a comunității academice. Dacă nu faci acest lucru, te excluzi din comunitatea academică.

Școala ne învață să evităm plagiatul în sensul tare al termenului. Nu copiați de la colegi! După cum este normal, pune mai puțin accent pe arta de a face trimiteri exacte în notele de subsol.

însușirea regulilor academice Așa cum este de așteptat, studenților și studenților din anii mici le vine greu să respecte regulile academice privitoare la citate și trimiteri, parafrizare și rezumare. Cred că persoanele care evaluează eseurile trebuie să ia lucrurile drept ceea ce sunt: un proces de inițiere treptată.

Oricât de dezamăgiți ar fi studenții și studentele, ei trebuie să știe că lucrările lor nu pot fi tratate așa cum sunt tratate studiile în sensul deplin al cuvântului. În perioada de învățare a regulilor comunității academice totul trebuie să decurgă treptat. Trebuie să dovedești că ai elaborat fișele bibliografice, că ai fișe de lectură, că ai lucrat la proiect și așa mai departe. Cu alte cuvinte, activitatea de elaborare a eseului este mai importantă decât produsul final.

Consolarea este că modul de a proceda prescris mai sus este și un mod de a proteja oameni tineri de un contact prea brutal cu regulile academice. De altfel, orice reguli trebuie însușite treptat. Există și aici o perioadă de tranziție.

Mi-ar fi imposibil să aduc dovezi serioase, dar experiența-mi sugerează că tiparul tranziției este pretutindeni același. Când am fost, la mijlocul anilor '90, la Oxford, eram foarte curios să aflu ce se petrece la meditații (*tutorials*). Evident, nu poți asista la o meditație așa cum asști la un curs. Doamna profesoară de filosofia științelor sociale a fost singura persoană de la care am putut afla câte ceva pe această temă. Studentele sau studenții trebuie să scrie săptămânal un eseu. Ce fac unii dintre ei, la început? Copie un text în bibliotecă. Ceva mai târziu traduc un text. Apoi încep să pună cap la cap note de lectură și așa mai departe. Oamenii sunt aceiași pretutindeni. Deprinderea regulilor jocului academic se face treptat. Avantajul secretului meditațiilor este totuși considerabil. Nu te simți pus sau pusă la stâlpul infamiei la fiecare pas al ritului de inițiere.

totul începe de la notele de lectură Care ar concluziile de ordin practic? Totul începe cu notele de lectură care disting scrupulos între parafrizare și extras. Un alt moment important este construirea unui proiect propriu. Citatele trebuie verificate cu atenție. Trebuie reexamineate, când este posi-

bil, sursele de bază, nu doar fișele de lectură sau fișele cu idei. De asemenea, rescrieți textul de cât mai multe ori. Probabilitatea de a scrie un text identic cu al altcuiva devine în acest fel neglijabilă.

6.1.2.5 Nu citați trunchiat

Există și reversul plagiatului. Extrasul este pus între ghilimele, trimiterea este absolut corectă, dar citatul este trunchiat.

Nu citați trunchiat. În mod voit sau accidental, s-ar putea să constrângeți textul să pară a spune ceea ce, de fapt, nu spune.

Să presupunem că un autor a scris „ca principiu metodologic, relativismul cultural este acceptabil“, iar noi am cita doar „relativismul cultural este acceptabil“. Textul a fost trunchiat în așa fel încât i-am atribuit persoanei care l-a scris altă idee.

6.1.2.6 Forma citatelor

Forma în care plasăm extrase din textele altora în eseu este importantă. Ea trebuie să sublinieze distincția netă dintre ceea ce au scris alții și ceea ce am scris noi.

Recomandarea noastră este ca citatele să fie plasate în subsol. Numai citatele deosebit de importante, cele care sunt obiectul unei analize speciale sunt plasate în textul de bază al eseului. Nu plasăm în textul de bază al eseului materialul care doar ilustrează sau completează cele scrise în textul de bază. Evităm astfel și impresia că susținerea celor afirmate de către noi vine din citate și nu din argumentele noastre.¹¹

Citatele scurte incluse în textul de bază se pun în ghilimele. Ele se încheie cu o trimitere la o notă de subsol în care este precizată sursa și locul din sursă de unde am extras citatul.

Citatele ample este absolut recomandabil să le separăm de textul de bază prin plasarea lor pe rânduri separate și retrase la ambele margini în raport cu textul de bază. În acest caz n-are rost să punem ghilimele. Modul în care este plasat textul arată că este vorba de un citat.

Citatul este precedat de un alineat care anunță într-un mod adecvat faptul că va urma un extras. Putem încheia acel alineat cu două puncte pentru a sublinia fără dubii că este vorba despre un citat.

Citatul se încheie cu o trimitere la o notă de subsol. Nota de

¹¹Nu recurgeți deci la faimoasa formulă „așa cum arată...“. Arătați chiar dumneavoastră de ce stau așa lucrurile

**puneți
citatele în
notele de
subsol**

**citate în
textul de
bază**

subsol începe fără *vide* (sau *v.*). Partea ei inițială arată exact de unde am extras citatul. Paginile sau indicația structurală trebuie să fie neapărat prezente.

Avantajul notei de subsol, față de o paranteză pusă sub citat, este acela că ea permite inserarea firească a unor comentarii suplimentare.

citate în notele de subsol În notele de subsol, citatele sunt introduse doar în ghilimele. Ele sunt urmate de o paranteză rotundă în care sunt incluse toate datele necesare pentru a identifica locul de unde am extras citatul.

ghilimelele franceze În cazul extraselor apar două probleme de punctuație. Prima privește cazul în care extrasul conține ghilimele. În eseurile scrise în limba română, dacă extrasul este pus în ghilimele, ghilimelele din extras se înlocuiesc cu ghilimele franceze.¹²

paranteze drepte În notele de subsol, parantezele rotunde în care sunt puse datele cu privire la publicare sunt înlocuite cu paranteze drepte. Această operație este executată pentru a evita confuziile cu parantezele rotunde în care sunt puse datele cu privire la locul de unde a fost extras citatul.

6.1.3 Evitarea limbajului imprecis

Limbajul imprecis este, de multe ori, apanajul ziarelor cărora în Anglia li se zice „tabloide”. Nu este nicidecum apanajul lor exclusiv. Acest limbaj pătrunde chiar și-n unele eseuri filosofice.

„Americani au clonat o oaie” – scrie cu litere mari ziarul. Dacă reluați o asemenea formulare în eseu dumneavoastră, folosiți un stil impropriu într-un eseu academic.

precizia academică Scrisul academic cere să adăugați la afirmația de mai sus o trimitere. Evident, ar fi imposibil să arătați către garajele sau atelierele tuturor americanilor. Trebuie să indicați o publicație anume, în care o anumită persoană prezintă un rezultat al cercetărilor sale. Prin urmare trebuie să reformulați totul într-un stil precis „X susține că a obținut următorul rezultat...“.

S-ar putea să aveți mari probleme cu găsirea unei surse pentru formularea precisă. În orice caz, nu-i trimiteți pe alții la textul din ziar. Folosiți ceea ce se spune concret în articol pentru a detecta o sursă adecvată.

¹²Ghilimelele franceze au forma «».

6.1.4 Nu excludem pe nimeni

Comunitatea academică este o comunitate deschisă. Oricine îi respectă regulile poate face parte din ea. Culoarea ochilor, lungimea nasului, vocea, faptul c-ai sau nu rude în străinătate și multe altele sunt irelevante.

De ce am exclude atunci pe cineva din publicul potențial? Dacă scriem un eseu academic, ne adresăm oricărei persoane care ar putea înțelege regulile jocului academic.

Expresiile pe care le folosim nu trebuie să sugereze că ne-am **evitați** adresa numai unui anumit grup. Nu contează aici dacă este vorba **limbajul** despre inițiații în tainele unei credințe sau despre persoanele care **discrimi-** doar se întâmplă să aparțină unui gen sau altul sau unei anumite **natoriu** etnii.

Cred că efectul pervers al folosirii unui limbaj discriminatoriu este acela că ne excludem pe noi din comunitatea academică. Ne excludem pentru că nu suntem în stare să-i respectăm caracterul deschis.

În practică, aceasta înseamnă că trebuie să ne ferim să folosim, de pildă, exclusiv un anumit gen. Nu toate persoanele care au scris un text filosofic sunt de gen masculin. Nu se cade deci să utilizăm doar cuvântul „autor“.

De multe ori este bine să folosim cuvinte sau expresii care se **folosiți** referă clar la ambele genuri. Putem să vorbim despre „persoane“ **ambele** sau despre „publicul cititor“. Alteori un „sau“ bine plasat rezolvă **genuri** problema. Dacă recursul la aceste soluții vi se pare supărător din punct de vedere stilistic, puteți echilibra pur și simplu referirile la genuri.¹³

În altă ordine de idei, citarea unor lucrări semnate de oameni ca Iosif Stalin sau Adolf Hitler nu reprezintă în sine o problemă. Există o sumedenie de situații în care lucrul acesta trebuie făcut, conform tuturor regulilor academice. În politica lor, Stalin și Hitler au judecat însă oamenii pe baza apartenenței lor la grupuri sociale sau etnice. Este o mare problemă dacă recurgem la acest mod de a gândi. Acest lucru se întâmplă, de pildă, dacă oferim drept explicație pentru punctul de vedere filosofic al cuiva o afirmație de genul

¹³ Cerințele prezentate aici sunt prezente în cele mai variate locuri. William C. Paxson[34, pp.243–246] formulează asemenea recomandări pentru cei care scriu în contextul lumii afacerilor. Daniel N. Fischel[11, 102–106] arată în detalii cum poate fi evitat limbajul rasist și sexist în scrierile cu caracter academic. Cei doi Lester[22, pp.181–182] arată cum trebuie revizuit textul lucrărilor studențești pentru a evita limbajul discriminatoriu.

„a fost x “, unde x indică apartenența la un grup (etnic, religios sau judecați de alt tip). În acest caz comitem o eroare: fie autorul sau autoa-ideile și rea discutată nici nu există ca persoane distincte de grup; fie aduc persoa- ceva nou și atunci n-are sens să fie înțeleși în baza provenienței lor nele pe dintr-un grup sau altul.¹⁴ Regula pozitivă ne spune că orice idee baza și orice persoană trebuie judecate pe baza meritelor și defectelor meritelor proprii.

La fel de inacceptabilă este o formulare de genul „aceasta nu-i filosofie, este doar o mostră de gândire x “. Dacă este o mostră de gândire x , atunci **trebuie** discutată tocmai pentru a putea trage o concluzie în legătură cu x .¹⁵

Multe persoane tind să complice discuția în legătură cu regulile din această secțiune. Ele invocă aici problema gravă a libertății cuvântului. Intuiția mea este că libertatea cuvântului trebuie să fie extrem de mare. Dar nu despre asta este vorba aici. Scrierile care încalcă regulile prezentate mai sus nu sunt scrieri academice. Sunt altceva. Pamflete, manifeste politice, orice vreți, dar nu scrieri academice. Trimiterile, citatele sau alte pretenții academice nu schimbă cu nimic situația.

6.1.5 Semnele de punctuație

Ca documentarist și redactor de editură am constatat că este mult mai ușor să găsești îndreptare sau dicționare ortografice ale limbii române decât este să găsești îndreptare de punctuație. Textul filosofic are însă mare nevoie de punctuație. O virgulă bine pusă elimină o ambiguitate care poate da naștere la discuții inutile.

Cine scrie un eseu filosofic are nevoie și de o înțelegere a fundamentelor normelor ortografice și de punctuație. Preferințele mele personale se îndreaptă către micul tratat al lui Al. Graur[15]. Cartea are 190 de mici secțiuni cu explicații foarte clare.

În munca practică, veți avea nevoie și de o lucrare specializată pe semnele de punctuație, în care sunt discutate numeroase exemple. Îndreptarul lui Sergiu Drincu[9] este o posibilă soluție.

S-ar putea să vă întrebați ce caută secțiunea despre semnele de

¹⁴ Antony Flew[13, §7.10-7.13] discută eroarea genetică: dacă x a evoluat din y , atunci x este în esență ca și y . Dacă x a evoluat, n-are cum fi la fel ca și y .

¹⁵ Antony Flew[13, §4.21] arată că acest gen de obiecții reprezintă o ilustrare a ocolirii vicioase a problemei aflate în discuție. Am presupus deja că știm cum stau lucrurile cu x , dar vrem să evaluăm ceva care constituie o parte a acestei gândiri.

punctuație sub cupola stilului eseului. În primul rând, după cum am arătat deja, „stil” nu înseamnă aici calofilie, ci exprimare exactă a ideilor. Din acest punct de vedere, semnele de punctuație au aceeași putere ca și cuvintele. Un semn de punctuație trebuie folosit pentru a indica în mod exact ce anume ați vrut să scrieți.

6.1.5.1 Punctul, punctul și virgula, semnul întrebării și semnul exclamării

Într-un eseu filosofic, punctul ar trebui să fie întâlnit cât mai des. **punctul** Propozițiile scurte contribuie la claritatea textului. Rescrierea lor este ușoară. Este mai lesne de controlat dacă din ele decurge sau nu concluzia pe care vrem să o susținem. Este mai facil de văzut dacă lipsește ceva în paragraf.

Să zicem însă c-am scris o propoziție compusă mult prea lungă. **punctul și virgula** S-ar putea să ni se pară o exagerare să o spargem în propoziții mai scurte. Putem recurge, în acest caz, la punct și virgulă. Efectul lungimii excesive este anihilat, dar subliniem și caracterul unitar al respectivei porțiuni de text.

După cum arată Al. Graur,¹⁶ nu se pune punct după titluri. Nu există nici un pericol de a confunda titlul cu restul textului.

Semnul întrebării, în schimb, cred că trebuie pus și-n titluri. În **semnul întrebării** scris, în limba română, fără semnul întrebării este greu de multe ori să-ți dai seama dacă este vorba despre o afirmație sau o întrogație.

Într-un eseu filosofic trebuie să existe cel puțin o întrogație la care încercăm să răspundem. Pentru a pune o problemă nu ajunge însă să folosim o propoziție de formă interogativă. Semnul întrebării poate chiar să lipsească. Mult mai important este să explicăm în ce constă întrogația avută în vedere. Această explicație trebuie adâncită pe baza unei analize a presuposițiilor întrebării.

Nu trebuie să recurgem în nici un caz la întrebări retorice. **evitați întrebările retorice** Acestea ar călca regula atât de importantă într-un eseu filosofic a susținerii cu grijă a oricărei idei. Cu alte cuvinte, semnul de întrebare trebuie tratat și ca o posibilă sursă de pericole.¹⁷

¹⁶Al. Graur[15, §18]

¹⁷Colin Allen crede că întrebările retorice subminează claritatea. Expun, de asemenea, eseul unor atacuri din partea celor care nu sunt de acord cu întrebarea retorică. Nu oferă însă nici un temei pentru persoanele care ar fi de acord cu ceea ce sugerează întrebarea retorică. Pentru detalii v. Colin Allen, *A Guide to Writing Philosophy Essays* [ghid pe situl Texas A&M University], <<http://grimpeur.tamu.edu/~colin/Phil416/writing.html>>, 2000–2002 (data accesării: 21/06/2003), „don't use rhetorical questions”.

semnul exclamării Semnul exclamării n-ar trebui să apară deloc în textul filosofic de bază. Al. Graur arată că acest semn se pune atunci când vrem să marcăm prezența afectivității. Utilizarea sa este la latitudinea persoanei care scrie. Dacă este pus un punct, atunci nu este vorba despre afectivitate. Dacă este pus semnul exclamării, autoarea sau autorul vor să adauge o nuanță afectivă.¹⁸ Eseul filosofic face însă apel la argumente, nu la afectivitate.¹⁹

Dacă vrem să formulăm o regulă, simplul fapt că verbul propoziției este la imperativ este suficient.

Cele spuse despre semnul exclamării se aplică doar textului de bază. Evident, dacă descriem o anumită situație sau cităm dintr-o operă literară și așa mai departe semnul exclamării s-ar putea să aibă rolul său firesc.

Considerațiile de mai sus explică motivele pentru care nici într-o serie dintre titluri din această carte nu apare semnul exclamării. În text ar fi apărut un punct la sfârșit, dar punctul nu se pune după titluri.

6.1.5.2 Virgula

Virgula este un instrument puternic. Mânuit prost poate să ne joace însă multe feste. Din fericire, textele filosofice, din punctul de vedere al scrisului, sunt mult mai ușor de realizat decât cele literare. Am selectat mai jos cazuri mai importante într-un text filosofic.

6.1.5.2.1 Condiționalele Se pune întotdeauna virgulă în propozițiile compuse după modelul „dacă... , atunci...“.

Cuvântul „atunci“ joacă un rol aparte în construcția de bază. Încercați să-l puneți pe „numai“ în fața lui „dacă“ și „atunci“ își va pierde pierdere rostul. Dar care este rostul lui „numai“ sau al lui „atunci“? Al. Graur are o teorie privitoare la mutarea accentului de pe o propoziție pe alta într-o propoziție compusă.²⁰ Aceste cuvinte marchează accentele puse de cine scrie.

Ce se întâmplă însă cu virgula atunci când se mută accentele? Sugestia lui Al. Graur este că virgula trebuie pusă când accentul cade pe propoziția principală. Când accentul cade pe propoziția

¹⁸ Al. Graur[15, §43]

¹⁹ În anexele acestei cărți veți întâlni semne de exclamare. Ele reflectă emoțiile utilizării/dezvoltării de programe. Gândiți-vă ce simte cineva care pierde fișierele în care este cartea pe care tocmai a scris-o!

²⁰ A se vedea Graur[15, §27].

secundară ea nu trebuie pusă. A se vedea diferența chiar în cazul regulii privind condiționala:

1. Dacă se pune accentul pe principală, atunci punem virgulă;
2. Numai dacă punem accentul pe condițională nu punem virgulă.

6.1.5.2.2 Explicație și descripție Explicațiile trebuie precedate de o virgulă. Scriem, de pildă, „filosoful X, care citise multe. . . , avea intuiții subtile în materie de. . . “. Propoziția pusă între virgule oferă un supliment de informații în legătură cu X, menit să pună într-o lumină mai clară faptul că X avea intuiții subtile într-un anumit domeniu.

Ce se întâmplă dacă ne întrebăm cine este X din exemplul de mai sus? Trebuie să furnizăm o precizare de genul „ea vorbea despre filosoful care a scris. . . “. Dacă punem virgulă înainte de „care“, atunci am trata lucrurile ca și cum am avea deja o descripție cu ajutorul căreia să-l identificăm pe X. Or, lucrurile nu stau așa.

Descripțiile reprezintă o problemă filosofică dificilă. Nu rezolvăm această problemă eliminând o virgulă. Regula este însă simplă: punem virgulă dacă adăugăm informații suplimentare, explicații; nu punem virgulă dacă formulăm o descripție.²¹

6.1.5.2.3 Concluziile Folosirea semnelor de punctuație diferă de la o limbă la alta. Câteodată, chiar pentru aceeași limbă, regulile pot să fie destul de ciudate. Așa stau lucrurile și-n cazul în care vrem să formulăm o concluzie.

În limba română, cuvinte sau sintagme precum „deci“, „prin urmare“, „așadar“ sunt semnele distinctive ale unei concluzii. Faptul că sunt marcajele care ne arată că propoziția reprezintă o concluzie ar trebuie să împiedice orice separare a lor de restul propoziției printr-o virgulă. Lucrurile stau numai în parte așa.

„Deci“ nu se desparte de restul propoziției prin virgulă.²² Nici **deci** dacă este pus în interiorul propoziției „deci“ nu se desparte de rest prin virgule.

În traducерile din limba engleză, multă lume cade în capcana mo-

²¹ Regula reprezintă o adaptare a formulării date, în termenii gramaticii tradiționale, de către Al. Graur[15, §26].

²² Drincu[9, §I.1.1.5] și Al. Graur[15, §35].

delului „*therefore/thus*, many philosophers believe that...“²³ Trebuie să traducem și ortografia, nu doar cuvintele.

Lingviștii români sunt de părere însă că „așadar“ și „prin urmare“ trebuie puse între virgule. Al. Graur apreciază că regula este prea tranșantă.²⁴ Și mie mi se pare c-ar trebui să fie permis ca atunci când vrem să scoatem în mod deosebit în evidență faptul că este vorba despre o concluzie să nu punem virgulă.²⁵

Regulile fiind așa cum sunt, cred că recomandabil ar fi să marcăm concluziile prin „deci“. Acest cuvânt nu trebuie despărțit de rest prin virgulă.

6.1.5.2.4 Atitudinile propoziționale „Atitudine propozițională“ este un termen tehnic filosofic pentru construcții de forma „cred că...“, „știu că...“. În aceste construcții, după „că“ stă o propoziție.

În cazul atitudinilor propoziționale, traducerea dintr-o limbă precum germana ne poate deruta. În limba germană, se desparte prin virgulă orice propoziție secundară.²⁶ Pentru atitudinile propoziționale, modelul este *X glaubt, daß p*.

În limba română NU se pune virgulă înainte de propozițiile introduse prin „că“ în atitudinile propoziționale.

6.1.5.2.5 Câteva reguli prohibitive Foarte multă lume are tendința de a plasa între virgule pe „însă“. În acest caz cădem în însă capcana întinsă de compunerea adversativă a propozițiilor. Propozițiile românești se compun adversativ după modelul „vine la curs, dar/însă nu ia notițe“. Cuvintele „dar“ sau „însă“ pot fi percepute și ca niște marcaje. Faptul că „însă“ se poate deplasa prin propoziție nu justifică punerea sa între ghilimele.

Considerațiile de mai sus se aplică și-n cazul lui „totuși“.

Un bun test al corectitudinii plasării unei porțiuni de text între virgule este eliminarea ei. Ce se pierde prin eliminare? Dacă am pierdut doar o precizare, o explicație suplimentară, atunci putem

²³ Am adaptat exemplul după cel din Dieter Giering et al., *English Grammar: A University Handbook* (Leipzig: Verlag Enzyklopädie, 1987), §7.8.4.

²⁴ Al. Graur[15, §35]. Caietul cu *Normele de editare Humanitas* de la începutul anilor '90 atrăgea în mod special atenția asupra faptului că „deci“ nu se separă prin virgule, dar „așadar“ se separă.

²⁵ Apelul la pauzele pe care le facem atunci când vorbim mi se pare fără sens. Semnele de punctuație sunt un sprijin în organizarea logică a textului. Organizarea logică a textului și efectele oratorice sunt două lucruri diferite.

²⁶ V. discuția foarte pertinentă din Al. Graur[15, §24].

păstra virgulele. Dacă se schimbă sensul a ceea ce vrem să comunicăm trebuie să eliminăm virgulele.

În cazul lui „însă” sensul se schimbă dacă scriu „vine la curs, nu ia notițe”. Construcția devine ambiguă. Ar trebui să precizez sensul textului. Pot adăuga ceva de genul „dar reține totul” sau exact opusul acestei susțineri. Deci „însă” își are rostul.²⁷

Cazurile compunerilor conjunctive sau disjunctive sunt prea cunoscute pentru a le trata aici.

6.1.6 Expresiile latinești

Trimiterile din textele academice folosesc o serie de cuvinte și expresii din limba latină. Lista se limitează doar la expresiile mai des utilizate în textele românești.

ap. *apud*(citat de). Se spune atunci când am citit ceva scris de X în opera lui Y. Trebuie folosit distinct de *vide*.²⁸

cf. *confer* (compară). Acest termen este folosit adesea în mod greșit. Trebuie distins cu grijă de *vide*.

et al. *et alii* (și alții). Se folosește atunci când menționăm doar o parte dintre autorii sau autoarele unui text. În acest caz, trebuie să arătăm că există și alți autori sau autoare.

etc. *et cetera* (și așa mai departe).

ibid. *ibidem* (în același loc). Se folosește atunci când trimitem la același autor și la aceeași lucrare.

id. *idem* (același). Se folosește în locul numelui autorului sau autoarei atunci când, în aceeași notă, menționăm mai multe scrieri ale sale.

in (în). Precede sursa unui document.

inf. *infra* (mai jos).

²⁷Nu vreau să insinuez prin exemplul de mai sus că trebuie luate neapărat notițe. Wittgenstein le cerea studentelor și studenților săi să nu ia notițe (v. Mircea Flonta, „Notă istorică” în Ludwig Wittgenstein, *Cercetări filozofice*(București: Humanitas, 2004), p.16). Exemplul vrea doar să sugereze cât este de importantă virgula în text.

²⁸Pentru mai multe explicații v. pagina 89.

loc.cit. *loco citato* (în locul menționat). Se folosește după numele autorului, când trimitem la lucrarea sa la care ne-am referit ultima oară.

m.m. *mutatis mutandis* (după ce am făcut schimbările necesare).

MS *manuscriptum* (manuscris). La plural, *manuscripta*; prescurtat, MSS.

op.cit. *opere citato* (în lucrarea menționată). Se folosește ca și *loc. cit.*

pass. *passim* (peste tot). Se pune atunci când ideea se întâlnește peste tot în lucrarea citată și nu vrem să ne referim doar la o anume pagină.

s.a. *sine anno* (fără an). Se pune atunci când nu putem identifica anul publicării textului.

s.l. *sine loco* (fără loc). Se pune atunci când nu putem identifica localitatea unde a fost publicat textul.

sup. *supra* (mai sus).

s.v. *sub verbo, sub voce* (la cuvântul). Se folosește atunci când facem o trimitere la un dicționar.

v. *vide* (vezi). Este modul cel mai des folosit de a trimite la un document.

Recomandarea noastră este ca utilizările lui *vide*, *apud* și *in* să fie diferențiate în mod strict. Punem *apud* doar când este vorba despre citarea prin intermediar a unui fragment sau a unor idei. Punem *in* numai pentru a marca începerea descrierii unui volum în care se află un document (articol, studiu, notă etc.). Punem *vide* în alte situații. n-are rost să punem *vide* la începutul notei dacă indicăm locul unde se află un extras sau o idee pe care le-am inclus în textul nostru; în acest caz, punem direct trimiterea ca atare. n-are sens să punem *vide* în paranteza care încheie textul citat în note.

Confer este sursa celor mai multe confuzii. Se folosește doar când vrem să facem o comparație. Altfel, trebuie pus *vide*.

Comparațiile trebuie făcute doar între alți autori, nu între scrierile noastre și ale altora. A folosi „așa cum scrie și...”, când este vorba de tine, frizează încercarea de a induce în eroare.²⁹ Trebuie

²⁹Mulțumesc colegului Marin Bălan pentru că mi-a atras atenția asupra acestui gen de eroare.

folosit *vide* dacă ideea coincide cu ceea ce a scris deja altcineva. Altfel, trebuie indicate diferențele.

Loc. cit. și *op. cit.* ar trebui evitate.³⁰ Dacă le folosim, trebuie să figureze în note foarte apropiate între ele.³¹

O serie de alte cuvinte și expresii latinești se întâlnesc frecvent în textele academice:

ad fontes (la surse). Se utilizează atunci când vrem să subliniem importanța surselor primare, a originalului în raport cu comentariile, dicționarele, traducerile sau orice alt fel de sursă secundară.

ad hoc (special pentru cazul dat). Se recurge la această expresie mai ales când vorbim despre o soluție care n-are un caracter general. Are uneori o nuanță peiorativă (presupoziția fiind că rezolvările trebuie un caracter general).

a prima facie (la prima vedere). Se spune despre ceea ce am înțeles la o primă lectură.

sic (așa este scris). Se folosește când vrem să atragem atenția că nu este vorba despre o greșeală de transcriere, ci așa este scris în textul original.

N.B. *nota bene* (observați cu atenție).

non seq. *non sequitur* (nu decurge logic). Se folosește atunci când în text întâlnim o inferență care nu este validă.

PS *post scriptum* (postscriptum). Porțiune de text adăugată după ce lucrarea a fost încheiată.

Q.E.D. *quod erat demonstrandum* (ceea ce trebuia demonstrat). Expresia este folosită pentru a marca terminarea unei demonstrații.

vs. *versus* (contra). Se folosește când vrem să indicăm un contrast. Apare în denumirile cazurilor din dreptul jurisprudențial.

³⁰ A se vedea aici explicațiile de la pagina 68.

³¹ Vedeți aici exemplul de la pagina 122.

6.1.7 Termenii limbii grecești vechi

Este inutil să explicăm pe larg ce loc important ocupă termenii grecești în scrierile filosofice. Cum ar trebui să cităm însă acești termeni?

Unii termeni au prins rădăcini în limba română. Este chiar cazul cuvântului „filosofie“. Ar fi ridicol să-i cităm în forma lor din greaca veche.

Există o serie întreagă de situații în care cuvântul grecesc, atunci când analizăm textele filosofice grecești, este mai bine să fie redat ca atare, în forma sa veche. Recurgem sau nu la transcripție?

Testul la care invităm mereu în capitolul de față este acela de a ne pune în locul publicului cititor. S-ar părea că lectura este mai ușoară dacă folosim transcripții. Cred că lucrurile nu stau așa. În primul rând, este nu este vorba despre publicul larg. În al doilea rând, facem loc erorilor.

Există cel puțin un exemplu de greșeală pricinuită de imprecizia transcrierii care a primit chiar și binecuvântarea dicționarelor. În mod normal, ar trebui să distingem între „autarhie“ și „autarchie“, dar nu facem acest lucru. Primul cuvânt înseamnă *putere absolută*; al doilea are sensul de *suficient sieși*.³²

Noi pledăm pentru evitarea transcripțiilor. Pentru a vedea modul concret în care se pot insera fragmente de text în greaca veche folosind L^AT_EX trebuie consultată anexa B.3.5.

6.1.8 Cuvintele înșelătoare

Cuvinte care sună normal în alte limbi pot să dea un aer ridicol unei propoziții românești. Auzi uneori vorbindu-se despre „munci penibile“. Probabil la mijloc este influența expresiei franțuzești *travail pénible*, care înseamnă *muncă anevoioasă, grea*.

În mediul universitar auzi multe persoane spunând că „aplică pentru o bursă“. Aici influența vine din engleză. Normal ar fi să zicem „fac cerere de bursă“.

În lumea calculatoarelor, sunt tot felul de expresii care contribuie vizibil la stricarea limbii române. Se vorbește despre programe care

³²Iată explicația unei specialiste în lingvistică: „ar trebui să se deosebească în scris și-n pronunțare elementele de compunere -*arhie* «conducere» și -*archie* «mulțumire, suficientă»; impunerea formației greșite *autarhie* în locul celei corecte, *autarchie*, denotă confuzia produsă de interpretarea grafiei ambigue“ (Mioara Avram, *Ortografie pentru toți* [București: Editura Academiei Române, 1990], p.32).

„suportă“. Cuvântul englezesc aflat în mintea celor care se exprimă astfel înseamnă *sprijin*. În românește sună hilar când este folosit în contexte nepotrivite. De multe ori, este vorba despre ideea simplă că programul este conform cu un standard sau permite folosirea unui limbaj și așa mai departe.

Cum se întâmplă de multe ori, unele erori flagrante au prins rădăcini solide. Spunem că „salvăm“ fișierul pe disc. Metafora din limba engleză este, în realitate, mult mai puțin patetică. Aluzia este la faptul că „punem ceva la păstrare“ (conservăm).

Se poate că în cazul „salvării“ intervine și reticența în fața metaforelor în stil englezesc. Limbajul nostru tehnic tinde să fie mult mai abstract. Nimeni nu zice „șoarece“, ci *mouse*.

Tendința către folosirea termenilor savanți conduce uneori la diferențe de nuanță care pot fi însă foarte înșelătoare în filosofie. Pentru multă lume termenul englezesc *mind* trebuie redat prin „spirit“, **spirit vs. minte** în limba română. În franceză nu există altă soluție, dar noi avem „minte“. ³³ Termenul acesta este, ca și *mind*, un cuvânt al limbii obișnuite și așa trebuie să și fie când ne referim la o serie întreagă de autori care au scris în limba engleză. ³⁴

Vorbele alunecă ușor dintr-o limbă într-alta. S-ar putea ca uneori să nu existe, în limbajul de specialitate, nici o altă soluție decât preluarea unui termen din altă limbă. Operația aceasta trebuie făcută însă cu mare precauție.

6.1.9 Exprimarea distincțiilor filosofice

Abundența distincțiilor conceptuale constituie o trăsătură frapantă a textelor filosofice. Este oare întotdeauna nevoie de forjarea unor cuvinte speciale pentru a exprima aceste distincții?

În excelenta sa monografie consacrată terminologiei filosofice românești, Ioan Oprea distinge patru căi de extindere a terminologiei filosofice:

- traducerea (folosind resursele limbii române obișnuite);
- crearea de noi unități lexicale prin calc total sau parțial;

³³După cum arată pe larg Ioan Oprea[31, §122] termenul „minte“ a fost amplu folosit în textele filosofice românești.

³⁴Este bine să ne uităm atent și la alte limbi romanice, nu doar la franceză, atunci când traducem din engleză. Clasică scriere a lui Bertrand Russell, *The Analysis of Mind*, este tradusă în italiană cu titlul *L'Analisi della mente* (Roma: Newton Compton Editori, 1969).

- crearea de noi unități lexicale prin împrumut;
- introducerea de noi înțelesuri.³⁵

În practică nu este foarte simplu de luat o decizie. Cred că procedurile trebuie aplicate în ordinea de mai sus. Am să ofer un exemplu care dă multă bătaie de cap.

Exemplul este foarte cunoscut și poate fi întâlnit în multe manuale de logică și filosofie. Să zicem că discutăm despre propoziția „Plouă.”; textul acestei propoziții este altul în engleză sau franceză decât în română; conținutul ei rămâne însă același. Dacă luăm însă drept exemplu o propoziție de genul „Mă dor dinții.”, lucrurile se schimbă în interiorul limbii române. Textul rămâne același, dar conținutul diferă, atunci când X și Y alcătuiesc scrisori în care includ această propoziție. Evident, situația este similară, *mutatis mutandis*, dacă avem în vedere contextele în care propoziția respectivă este rostită, nu scrisă.

În limba engleză se folosește cuplul *sentence-proposition* pentru a exprima această distincție. Ce putem face oare cu resursele de care dispunem în limba română?

Cuvântul „propoziție” este un neologism în limba română. Este limpede c-a fost introdus pentru că nu aveam cum traduce termenul respectiv folosind resursele limbii mai vechi. N-avem deci cum recurge la traducere.

Calcul nu este nici el de mare ajutor aici. Dar împrumutul? Cuvântul englezesc *sentence* provine din latinescul *sententia*.³⁶ Prin intermediul filierei franceze, noi avem însă în limbă latinescul *sententia* sub forma cuvântului „sentință”.³⁷ N-avem deci nimic de împrumutat.

Cuvântul „sentință” are, în românește, înțelesuri foarte clare: *decizie judecătorească, gând exprimat concis, aforism*.³⁸ Este practic imposibil să-l încercăm și cu înțelesul de *text al unei propoziții* sau *rostită a unei propoziții*.

Urmând modelul limbii franceze, unde *phrase* înseamnă *unitate care exprimă mesajul persoanei care vorbește* și unde putem vorbi despre *une phrase simple*, compusă dintr-o sintagmă nominală și un

³⁵ Oprea[31, p.225].

³⁶ Walter W. Skeat, *A Concise Etymological Dictionary of the English Language* (Oxford: Clarendon Press, 1882), s.v. După Skeat, înțelesul cuvântului latinesc este cel de *mod de a gândi*.

³⁷ *Dicționarul limbii române* (București: Editura Academiei Române, 1990), vol.X, partea a 3-a, s.v.

³⁸ *Ibidem*.

verb,³⁹ am putea folosi „frază” pentru *sentence*. Soluția nu mi se pare fericită. La noi, sub influența gramaticilor de școală, „frază” are limpede înțelesul de *propoziție compusă*.

Aceeași problemă o ridică folosirea termenului „judecată” pentru *proposition*. Este adevărat că textul unei propoziții exprimă o judecată. Problema este că „judecată” este un termen al logicii clasice. Aici el este încărcat cu încă un înțeles.

Dacă recurgem frecvent la introducerea de noi înțelesuri, atunci supraîncărcăm textul cu termeni ambigui. Înlăturarea ambiguității permanent prezente în text cere un efort la lectură. Un asemenea text este foarte vulnerabil la o lectură grăbită. Concluzia noastră este că a patra procedură de creare a terminologiei filosofice nu trebuie folosită în mod forțat.

Ce ne rămâne de făcut? Trebuie să examinăm spiritul limbii române. Limba română nu se împacă ușor cu formarea de cuvinte compuse. „Gâtlegăul” nu este acceptabil. Nu era acceptabil în secolul al XIX-lea și n-ar fi acceptabil nici azi ca un calc, să zicem, după englezescul *necktie*. Spre deosebire de franceză, suntem însă mai toleranți la împrumuturi. Așa că folosim un termen precum „cravată”, împrumutat din limba franceză.

Când nu putem împrumuta, recurgem la sintagme. Sugestia noastră ar fi deci să spunem simplu „textul propoziției” și „conținutul propoziției” când vrem să redăm distincția dintre *sentence* și *proposition*.

Când nu suntem atât de preocupate sau preocupați de distincția dintre textul și conținutul propoziției folosim doar „propoziție”. În logica formală a propozițiilor, de pildă, avem în vedere doar tehnici de manipulare corectă a textului propozițiilor. Vrem să știm, de fapt, când putem rescrie premisele sub forma unei concluzii (unor concluzii) în așa fel încât ori de câte ori premisele sunt adevărate concluzia să fie și ea adevărată. În limba română, arta scrisului este cea a ambiguității bine temperate: atunci când nu se produc confuzii periculoase folosim termeni ambigui; când există pericolul unor neînțelegeri dramatice recurgem la termeni sau sintagme precise.

ambiguitatea bine temperată

Multă vreme m-am ferit de redarea termenului englezesc *belief* prin „credință”. Problema părea să fie pierderea distincțiilor în raport cu *faith*. Soluția este foarte simplă. Dacă se produc confuzii periculoase, atunci recurgem la sintagma „credință religioasă”. N-are însă românescul „credință” conotații exclusiv religioase? Nici vorbă.

³⁹Jean Dubois et al., *Dictionnaire du français contemporain* (Paris: Larousse, 1966), s.v.

Nu spunem oare despre cineva că este de bună credință? Nu este nici un sens religios aici. De asemenea, dacă mă urc în trenul de metrou care merge către Militari, dar socotesc că mă aflu în cel care merge către Gara de Nord, pot foarte bine spune că aveam credința (falsă) că mă îndrept spre Gară. „Credință“, folosit după ideea ambiguității bine temperate, este un termen cât se poate de bun.

N-ar trebui să ne jeneze faptul că avem o limbă cu atât de puține cuvinte? Întrebarea are o presuposiție eronată. Prezența multor cuvinte nu este neapărat un semn de rafinament cultural. Ea poate să reflecte uneori un stil diferit de formare a cuvintelor. Alteori poate să dea mai degrabă o notă de primitivism. Cuvintele multe pot să fie și un simptom al lipsei capacității de generalizare.

Limbile naturale evoluat au puteri echivalente atunci când este vorba despre exprimarea distincțiilor conceptuale. Stilul în care fac acest lucru diferă. Greșită este încercarea de a folosi un stil nepotrivit cu limba în care scrii. Rezultatul ei este un eseu cu pretenții savante, dar lipsit de claritate.

6.2 Corectura

Corectura este o profesie, așa cum o profesie este și munca de redactor. Misiunea corecturii profesionale dintr-o editură este foarte clară: ea trebuie să asigure concordanța dintre textul trimis de către editură la tipografie și textul realizat în tipografie. Operația este migăloasă. Trebuie verificate, de pildă, și despărțirile în silabe sau felul în care au fost tratate ortogramele. În practică, apar și diverse sugestii referitoare la ortografie. Constrângerea este însă limpede: nu poți adăga sau scoate nici un cuvânt, nici un semn din textul prelucrat în editură.

Din păcate, eseurile studențești, de la simpla lucrare de seminar până la lucrările de diplomă și disertațiile de master, trebuie redactate și corectate de către autoare sau autor.

De ce „din păcate“? Pentru că este greu să găsești redactor sau corector mai prost decât autorul unui text. Am simțit acest lucru și pe propria mea piele. Autorul tinde să citească textul din minte, nu pe cel de pe hârtie. Trece cu nepăsare peste erori pe care le-ar vedea fără probleme în textele altora.

Soluția pe care o recomand, în cazul corecturii ortografice, este recursul la un program de calculator. Ea trebuie completată cu folosirea unui sistem, precum cel descris pe larg în anexe, în care se face o separare între ceea ce vedeți pe ecran când scrieți și ceea ce

vedeți în pagîna tipărită. Faptul că pagina tipărită sau vizualizată pe ecran arată altfel decât manuscrisul este un avantaj colosal. Ea recrează ceva din distanța care există între ceea ce este în mintea ta și ceea ce vezi în textele scrise de către alții.

Există și un avantaj al corecturii făcute de către autoare sau autor. Aceștia pot face mici schimbări în text, dacă ele contribuie la îmbunătățirea aspectului grafic al paginii.⁴⁰ Lucrul acesta poate fi făcut numai de către autoare sau autor.

6.2.1 Literele

Există o regulă extrem de simplă pe care o știe oricine a corectat profesional texte. Nu căuta înțelesurile textului, vezi care sunt literele folosite!

Există și un alt principiu important în materie de greșeli de literă și de greșeli tipografice în general. Erorile tipografice se împart în două categorii: cele pe care le poate corecta publicul cititor și cele care nu pot fi corectate de către publicul cititor.

erori tipografice

Dacă în text apare scris „filsofie“, eroarea tipografică denotă neglijență, dar nu va împiedica pe nimeni să înțeleagă despre ce este vorba.

Eliminați însă cu grijă erorile care dau naștere unor formulări care nu pot fi îndreptate decât dacă știi ce a vrut să spună persoana care a scris textul.

6.2.2 Ortogramele

Când scriem este firesc să fim atente și atenți la moleculele textului, propozițiile, și la structurile formate cu ajutorul acestor molecule. Ne îndreptăm mai puțin atenția către structurile atomice ale textului.

Orice text are însă în compoziția sa elementele minime care susțin înțelesurile sau rolurile gramaticale. Când grafia acestor elemente (cuvinte și sintagme) este corectă se vorbește despre „ortograme“.⁴¹

Unele erori ar putea fi îndreptate de către publicul cititor. Este însă jenant să găsești scris, într-un eseu filosofic, „ideie“ în loc de „idee“.

⁴⁰Eu elimit, de exemplu, în acest fel *bad box*-urile semnalate de către L^AT_EX.

⁴¹V. definiția dată de Iancu și Horvat[19, p.8]. Pentru ei, ortogramele constituie obiectul de studiu al ortografiei.

„Crează“ în loc de „creează“ nu reprezintă desigur o catastrofă filosofică, dar eroarea ar trebui corectată.

Supărătoare este tendința de a scrie separat cuvinte care se rostesc împreună. Aici limba română are regulile ei. Se scrie „c-ar“, nu „că ar“.⁴² Grafia românească este subtilă: „c-ar“ se opune lui „car“.

Este posibil să evităm, în unele cazuri, elidarea.⁴³ Putem să nu elidăm, de pildă, pe „dacă“. Cred că trebuie exploatată această posibilitate doar pentru a sublinia rostirea separată. S-ar putea, din rațiuni stilistice, să dorim acest lucru: este un semn că porțiunea respectivă trebuie citită cu mai multă atenție.

Trebuie să fim atenți când corectăm cu programe de computer. Aceste programe pot da numai semnale orientative atunci când este vorba de „c-ar“ și „car“. Aici trebuie să intervină judecata umană.

filozofie Am lăsat intenționat la urmă discuția despre „filosofie“ și „fi-
vs. lozofie“. Al. Graur arată, pe bună dreptate, c-ar trebui să scriem
filosofie „filozofie“.⁴⁴ De ce n-am scrie „poésie“ și nu „poezie“? „Filosofie“ nu este o ortogramă din perspectiva limbii române.

De ce scriu totuși „filosofie“? Este un exemplu perfect de eroare cu care nu merită să pierzi vremea. Așa scrie pe frontispiciul facultății de specialitate a Universității din București. Nu se câștigă nimic dacă se corectează o inconsecvență ortografică minoră, care, la urma urmei, sugerează o ascendență atât de nobilă. Aici este efectiv suverană regula care ne cere să ne uităm cu ochii altora la textul nostru și să aducem textul la o formă curată din punctul lor de vedere.

6.2.3 Semantica textului

Înțelesul multor cuvinte din limba română poate să ne joace feste. De cele mai multe ori nu se produce nici un dezastru din punct de vedere filosofic. Ochii care se uită grăbiți peste eseu s-ar putea să se

⁴²V. în cartea lui Iancu și Horvat[19, p.73].

⁴³V. ce explică Iancu și Horvat[19, p.145].

⁴⁴V. argumentele relevante în Al. Graur[15, §166]. În ciuda a ceea ce pare a fi o inexactitate la Graur (cuvântul ὁ ζόφος [obscuritate] se găsește totuși în greacă în toată lunga ei istorie), ar fi oricum greu să percepem vreo legătură între „filozof“ și ζόφος. Sunt multe cuvinte compuse cu φίλο- în greacă, dar nici unul cu ζόφος sau un derivat al său. Émile Personneaux, *Dictionnaire grec-français* (Paris: Belin, 1925), pp.1519–1526, care consemnează numeroase asemenea cuvinte, nu oferă nimic care să poată genera confuzii. Nici alte dicționare pe care le-am consultat nu conțin un asemenea cuvânt compus. Oricum, cine ar putea să găsească un asemenea cuvânt compus sau să perceapă o ambiguitate ar ști suficient de multă greacă pentru a înțelege sensul cuvântului „filozofie“. În concluzie, nu există nici o nuanță care ar fi câștigată prin distincția „filosofie“ vs. „filozofie“.

îndrepte tocmai asupra unor pasaje în care folosim cu stângăcie un cuvânt. Efectul este mai degrabă unul neplăcut.

Să luăm drept exemplu cazul cuvintelor „nivel“ și „nivelă“. Filosofii și filologii nu prea au pe acasă o nivelă. Aceasta este o unealtă cu care zidarii verifică orizontalitatea sau verticalitatea suprafețelor pe care le construiesc. Cuvântul „nivelă“ provine din franțuzescul *nivelle*⁴⁵.

Termenul „nivel“ este, în schimb, folosit adesea în textele filosofice, cu sensul de *stadiu, grad*. Provine tot din limba franceză, dar din cuvântul *niveau*, adaptat la cerințele fonetice ale limbii române.⁴⁶

Cele două cuvinte au la plural forme diferite: „nivele“ pentru „nivelă“, „niveluri“ pentru „nivel“. „Nivelele“ sunt uneltele zidarilor. „Nivelurile“ se referă la diferențe dintre stadii, planuri, grade ale unor procese sau obiecte.

Nu este greu să repari, cu ajutorul contextului, eroarea produsă de utilizarea eronată a termenului „nivele“. Acest lucru cere un efort suplimentar din partea publicului cititor. Oricât de mic ar fi acest efort, el tot se simte la o lectură grăbită și afectează capacitatea eseului de a **supraviețui** unei asemenea lecturi.

Limba română întinde o sumedenie de asemenea capcane semantice. Pentru evitarea lor este bine să consultăm lucrări specializate.⁴⁷

6.2.4 Gramatica

Este foarte puțin probabil să găsim într-un eseu academic pe „este“ în locul lui „sunt“. Există însă și erori mai subtile. Mulți dintre noi s-ar putea să avem tendința să le comitem.

Un caz pe care-l putem întâlni în multe locuri este cel al verbului „a trebui“. Acest verb se folosește, de regulă, doar la persoana a treia singular.⁴⁸ Cu alte cuvinte, este un verb impersonal.

În textele filosofice cred că este absolut recomandabil să-l folosim pe „trebuie“ strict impersonal. Nu există deci nici un fel de „trebuie“ etc. în acest context.

⁴⁵Pentru etimologia cuvântului „nivelă“ v. Florin Marcu, *Noul dicționar de neologisme* (București: Editura Academiei Române, 1997), s.v.

⁴⁶Pentru etimologia cuvântului „nivel“ v. Alexandru Ciorănescu, *Dicționarul etimologic al limbii române* (București: Editura SAECULUM I.O., 2002), s.v.

⁴⁷O asemenea lucrare este cea a lui Silviu Constantinescu, *Dificultăți semantice* (București: Editura Științifică, 1994).

⁴⁸Alf Lombard și Constantin Gădei, *Dictionnaire morphologique de la langue roumaine* (București: Editura Academiei, 1981), s.v. au o formulare prudentă pentru regula care guvernează acest verb: *3^e pers. surtout*.

Gramatica eronată poate fi îndreptată de către publicul cititor. Funcționează însă, din nou, principiul supraviețuirii la o lectură grăbită. Erorile de orice fel solicită un efort suplimentar pentru a putea înțelege textul. Acest efort nu este în mod cert făcut de către o persoană care trebuie să parcurgă rapid zeci sau chiar sute de eseuri.

6.3 Computerul și faza finală a elaborării textului

Marele avantaj al programelor de computer specializate în procesarea textului este că fac totul din mers. N-aveți nevoie de un efort special pentru ca aparatul auxiliar al textului să arate perfect.

Există o operație computerizată care este mai bine să fie făcută la sfârșit: verificarea corectitudinii ortografice a textului. Tehnic este posibil să faceți această operație din mers. Noi pledăm însă pentru separarea ei de elaborarea textului ca atare. Operația este departe de a fi în întregime mecanică. Un cuvânt precum „in“ este corect ortografiat sau greșit ortografiat în funcție de context: profesorul era *în* școală; pictorul lucra cu ulei de *in*.

Corectarea textelor este profesie. Evident, doar editurile își permit să plătească pe cineva pentru această operație. Autoarele sau autorii sunt corectori mizerabili când este vorba de propriile texte.

Studiați anexa A.5 pentru a vedea cum găsiți în computer un ajutor de nădejde când este vorba de corectură.

Capitolul 7

Dincolo de elaborarea eseului: apărarea și evaluarea eseului

Cuprins

7.1	Susținerea eseului	113
7.2	Evaluarea eseului	114
7.2.1	Evaluarea eseului și neutralitatea acade- mică	116
7.3	Eseuri trimise altor universități	117
7.4	Computerul și susținerea eseului	118

În mod inevitabil viața de sine stătătoare a eseului se concretizează într-o varietate de interpretări ale textului său. Diversitatea lor n-ar trebui să deranjeze pe nimeni. Din contră, ar trebui să fie un prilej de bucurie.

Dacă este să formulez o recomandare, înainte de a discuta despre susținerea eseului, atunci aceasta ar fi aceea de a evita cu grijă afirmațiile de genul „aceasta este cea mai bună carte pe care am scris-o“ sau, la extrema cealaltă, „cred că această lucrare este slabă, n-am reușit să...“.

Lăsați publicul cititor să judece! Scrieți o lucrare ușor de parcurs și de evaluat și nu vă bateți capul cu evaluarea ei.

7.1 Susținerea eseului

Susținerea eseului (în fața unei comisii, în fața colegilor la seminar sau în cadrul unei conferințe) are o regulă extrem de simplă, dar pe

prezențați proiectul care multă lume o ignoră. În cadrul susținerii trebuie prezentat pe scurt modul în care a fost dezvoltat proiectul. Nu prezentăm textul, ci proiectul.

Textul poate fi citit de către oricine. N-are rost să-l citim. Dacă socotim că publicul n-are timp să parcurgă textul în întregime, redactăm un rezumat. Oferim acest rezumat spre lectură.

Proiectul este, în schimb, prin natura sa invizibil. Comisiile de examen vor chiar să se convingă că el a existat cu adevărat.

De altfel, nu este deloc dificil să explicați care au fost obiectivele proiectului, ce probleme ați atacat. Este important să arătați de ce v-ați oprit asupra unei soluții sau a alteia. Nu uitați să precizați limpede aria problemelor abordate și care dintre problemele cu care v-ați confruntat continuă să rămână o problemă deschisă. Nimeni nu se poate ocupa cu toate problemele dintr-un domeniu și cu atât mai puțin poate să le rezolve pe toate.

Împarantezarea este și ea o tehnică utilă în cazul susținerii. Tehnica împarantezării provine din lucrările cu caracter didactic. O lecție construită folosind împarantezarea începe cu o prezentare a obiectivelor (ce vor ști studentele și studenții după ce au studiat lecția), continuă cu corpul principal al lecției și se încheie cu un scurt rezumat care trebuie să corespundă punct cu punct obiectivelor anunțate.

Nu este greu să adaptați această tehnică la susținerea eseului. Corpul susținerii trebuie să fie însă mult mai concis decât cel al lecției. Ați obține contrariul efectului scontat dacă vă apucați să țineți o prelegere comisiei. Corpul susținerii este format din precizarea problemei abordate, a ipotezei de lucru și a modului în care ați analizat ipoteza și consecințele ei.

7.2 Evaluarea eseului

Din punctul de vedere al studenților și studentelor de la cursurile mele, mi se pare important să precizez modul în care văd evaluarea eseurilor. Criteriile prezentate în această secțiune nu trebuie însă sub nici o formă să fie percepute drept criterii împărtășite de către orice alt cadru didactic.

Experiența acumulată de-a lungul anilor îmi spune că este mai important să fie evaluată activitatea de elaborare a eseului și calitatea susținerii decât textul ca atare. Acest principiu explică și criteriile care urmează:

1. Calitatea proiectului (îndeosebi identificarea corectă a unei probleme existente efectiv în literatura de specialitate);
2. Bibliografia (identificarea dezbaterii din literatură);
3. Profunzimea cunoașterii literaturii (probată prin discuții, nu doar de către text);
4. Calitatea structurării interne a textului (împărțirea pe secțiuni; trecerile coerente de la o secțiune la alta);
5. Rescrierea textului (măsura în care textul a fost refăcut în urma unor reflecții critice);
6. Separarea netă a ideilor altora de ideile proprii;
7. Calitatea notelor de subsol, a trimiterilor interne (și a indicilor, în cazul eseurilor mai ample);
8. Capacitatea de a completa dezbaterea din literatură cu reflecții independente;
9. Calitatea susținerii;
10. Calitatea răspunsurilor la întrebările de la susținere.

Criteriile acestea (pe puncte) trebuie folosite echilibrat, în lumina a două principii:

- Acordarea unei bonificații pentru punctele acoperite cu prisosință;
- Penalizare necondiționată dacă un punct n-are o acoperire minimală.

Bonificațiile pot compensa satisfacerea parțială a cerințelor de la unul sau mai multe puncte. Ele nu pot fi folosite acolo unde un criteriu este complet neîndeplinit. Altfel spus, nu se operează o însumare aritmetică simplă a punctelor sau fracțiunilor de punct obținute. În felul acesta, persoanele care excelează din anumite puncte de vedere pot compensa slăbiciunile pe alte dimensiuni. Nu pot însă compensa ignorarea cerințelor minime. În acest caz se aplică o penalizare necondiționată.

Ideea de a ține cont de impresia de ansamblu mi se pare greșită. Așa cum arată și expresia folosită, este vorba doar despre o **impresie**. Nu există standarde clare pentru a putea transforma această impresie în puncte sau fracțiuni de puncte.

forma academică Forma academică a eseului are o importanță crucială. Dacă n-are o formă academică eseul ar trebui comentat în lucrări care au ele o formă academică. Nu se pune problema efectuării unei asemenea operații cu eseurile prezentate pentru a promova examene.

Conținutul depinde și el de o serie de constrângeri formale. Totul începe cu identificarea problemei. Dacă problema este inventată *ad hoc*, atunci cinci-șase criterii sunt încălcate automat. Nici nu mai ai, de pildă, cum separa ideile tale de ale altora. Reflecțiile completează o discuție care nu există și așa mai departe.¹

7.2.1 Evaluarea eseului și neutralitatea academică

Dacă vi se pare că standardele de evaluare sunt excesiv de orientate către forma eseului filosofic, trebuie să vă gândiți la dezavantajele alternativelor.

A evalua eseurile din perspectiva punctului de vedere adoptat înseamnă implicit a forța adoptarea de către studenți și studenți a punctului de vedere împărtășit de către persoanele care evaluează eseul. Acest mod de a proceda distruge, pe de o parte, capacitatea de a reflecta în mod independent. Pe de altă parte, descurajează credința că argumentele ar avea vreo importanță în filosofie.

Douglas Portmore arată că notele primite pentru eseuri trebuie să depindă de calitatea argumentelor oferite. Coincidența cu punctul de vedere prezentat la curs nu contează. După părerea sa, modul acesta de a proceda sporește șansele ca eseurile să fie scrise în mod inteligent.²

Acest tip de evaluare nu implică, pe de altă parte, ideea că persoanele care evaluează eseuri n-ar trebui să aibă un punct de vedere. Din contră, cred că acest stil de evaluare dezleagă, ca să zicem așa, mâinile cadrelor didactice. Acestea nu sunt nevoite să prezinte doar biografiile filosofilor și un rezumat cu caracter documentar al scrierilor acestora. Cadrele didactice pot, la rândul lor, să se angajeze în discutarea problemelor reale. La urma urmei, trebuie să dea un exemplu.

Secretul acestui tip de evaluare este legat, în fond, de ideea de

¹Evident, eu nu contest posibilitatea ca eseul filosofic respectiv să aibă o valoare literară sau de alt gen. S-ar putea chiar să aibă un public sensibil mai larg decât o scriere academică. El poate să dea glas, de pildă, unei stări de spirit a publicului. Toate lucrurile acestea au valoarea lor, dar în afara cadrului academic. Poate să sune cinic, dar este perfect posibil ca eseul în cauză să stea la baza a zeci de comentarii academice.

²Douglas Portmore[35, „adopting a position“].

neutralitate academică. Pasiunea pentru o idee nu este transformată în standard după care sunt judecate lucrările altora. În general, pasiunea nu ia locul argumentării. Valorile nu sunt lăsate să distorsioneze judecata.

7.3 Eseuri trimise altor universități

Principiile din această carte sunt utile oriunde în lumea academică de pe mapamond. Standardele diferă însă mult în detaliile lor. Cum detaliile contează de multe ori enorm, trebuie consultate manuale care prezintă cerințele universității la care este trimis eseu.

Experiența noastră în cazul universităților europene este foarte limitată. Ea este ceva mai bogată în cazul SUA, așa că referirile noastre concrete vor fi la universitățile din SUA.

Dacă participați la un concurs pentru obținerea unei burse în SUA, vă veți ciocni constant de cerința de a prezenta un eseu (*paper*). Nu uitați, înainte de toate, să supuneți unei revizuii atente cunoștințele dumneavoastră de limba engleză.³

Cerințele care privesc forma eseului diferă nu atât de la o universitate la alta, cât de la o comunitate academică la alta. Este bine să culegeți informații cât mai exacte despre standardele universității la care concurați.

Un sit cu foarte multe informații atât pe tema documentării, cât și a redactării lucrării este cel al Diane Hacker. Filosofia este clasată aici printre disciplinele umaniste și stilul recomandat pentru lucrări este MLA.⁴

Un alt sit care poate fi consultat cu folos este cel al Universității din Ohio. Puteți găsi acolo un set de ghiduri ale citării conform unora dintre stilurile consacrate în mediul universitar american.⁵

Stilul MLA (Modern Language Association) este prezentat în multe din manualele de elaborare a unei lucrări studentești.⁶ Pe

³Există cursuri de tehnica scrisului destinate persoanelor a căror limbă maternă nu este engleza. Cursul lui R.R.Jordan[20] este un exemplu în acest sens. El cuprinde prezentări ale unor tehnici esențiale pentru elaborarea unui text: realizarea unei descrieri, formularea unei definiții, introducerea de exemple, clasificarea, comparațiile, prezentarea legăturii dintre cauze și efecte, generalizare, interpretarea datelor, formularea de argumente.

⁴V. situl Diane Hacker[16, philosophy.htm].

⁵The Ohio State University, *Research and Internet Guides and Tools*, <<http://www.lib.ohio-state.edu/guides/>>, 2003 (data accesări: 29/02/2004).

⁶V., de pildă, Lester, Sr. și Lester, Jr.[22].

situl Diane Hacker găsiți și un model de lucrare care respectă forma cerută de către MLA.⁷

MLA are câteva particularități importante în raport cu recomandările din această carte. Toate lucrările citate trebuie trecute pe o listă finală, care se și numește *Works Cited*. Trimiterile la lista finală se fac într-o paranteză în care se pune numele de familie al autoarei sau autorului și pagina. Universitatea s-ar putea să vrea ca textul să arate ca pe vremea mașinilor de scris: spațiat la două rânduri, cu anumite margini etc. Toate cerințele privitoare la formă trebuie verificate până la cel mai mic detaliu. Nu trebuie neglijate în special detaliile referitoare la punctuația citatelor și a listei bibliografice.

În științele sociale, standardul este cel APA (American Psychological Association). Stilul APA folosește trimiteri de forma „nume, an, pagini“. Este destul de diferit de MLA.

Ambele stiluri admit citate plasate în textul eseului. Trebuie studiat însă cu atenție modul de așezare în pagină a citatelor.

În alte zone umaniste, cum ar fi istoria, s-ar putea să vă întâlniți cu stilul Universității din Chicago.⁸

Lucrările oferite drept model în multe manuale sau ghiduri de pe Internet trebuie studiate pentru forma lor. Nu trebuie uitat că modelele se adresează, de regulă, celor din anii mici. Sunt unele lucruri fundamentale care trebuie deprinse și-n planul conținutului din aceste lucrări. În special trebuie studiat modul de prezentare a unei literaturi, felul în care sunt surprinse liniile de forță ale dezbaterii.

7.4 Computerul și susținerea eseului

Sistemul \LaTeX , prezentat în anexe, permite și realizarea de diapozitive pentru computer. Acestea sunt aduse până la urmă la forma unui fișier pdf, care poate fi folosit pe orice computer conectat la un proiector. Diapozitivele sunt ideale pentru susținerea eseurilor, a conferințelor. Sunt eficiente și-n cazul prelegerilor.

Realizarea diapozitivelor computerizate nu este descrisă în anexe. Citiți documentația pachetului pdfslide din documentația distribuției MikTeX.⁹

⁷V. pe situl Diane Hacker[16, dalyper.pdf].

⁸Mai multe informații despre particularitățile diferitelor stiluri academice internaționale ne propunem să oferim pe situl cărții.

⁹Pentru instalarea distribuției MikTeX v. §B.1.2

Anexe

Anexa A

Creionul electronic

A.1 Sistemul de operare și managementul fișierelor

- 5 Nu este suficient ca un computer să fie în perfectă stare de funcționare ca simplu dispozitiv fizic. Fără un sistem de operare este un obiect decorativ. S-ar putea să fie util pentru a presa ceva sau pentru a împiedica accesul pisicii într-un ungher unde nu vreți ca ea să intre.
- 10 Ce este sistemul de operare? Ca și filosofia, sistemul de operare nu este ușor de definit. Tanenbaum și Woodhull spun că sistemul de operare este programul fundamental al computerului.¹ Ce face însă acest program? Am putea să-l vedem ca pe un program care creează o **mașină virtuală**. Aceași mașină fizică devine fie o mașină
- 15 Windows98 sau WindowsXP, fie o mașină Linux sau altceva. Am putea vedea sistemul de operare și ca pe un manager al resurselor sistemului.²

A.1.1 Mașina virtuală ideală

- Care mașină virtuală este mai bună? În ce să-mi transform computerul? Totul depinde de ceea ce vrem să facem cu computerul.
- 20

Dacă vrem să aflăm care este structura unui sistem de operare, atunci Minix sau Linux sunt sistemele de operare la care trebuie să

¹Tanenbaum și Woodhull, *Operating Systems* (Upper Saddle River, NJ: Prentice-Hall, 1997), p.1.

²Aceste posibilități de caracterizare a sistemului de operare sunt evidențiate de către Tanenbaum și Woodhull, *op.cit.*, pp.3–5.

apelăm. Sistemul Minix este descris de Tanenbaum și Woodhull.³ Prin contrast, sursele Windows sunt secrete. Firma care le realizează își protejează astfel drepturile sale de proprietate.

Dacă vrem doar să utilizăm un sistem de operare, atunci alte criterii intră în joc. Achiziția unei licențe pentru un sistem Windows98 sau pentru o versiune de casă a sistemului WindowsXP nu este chiar așa de costisitoare și costurile totale ale deținerii unui astfel de sistem sunt probabil sub cele ale unui sistem cu sursele deschise. Multe periferice, precum scannerile sau aparatele foto, se integrează foarte ușor sub Windows. Nu se poate spune același lucru despre Linux.

Cum cea mai mare parte a publicului care consultă acest ghid lucrează cu un sistem Windows, soluțiile descrise aici sunt cele disponibile pe acest sistem.⁴ Presupunerea noastră a fost una minimală. Am pornit de la ideea că sistemul utilizat este Windows98. De altfel, dacă nu sunteți conectați direct la Internet și prelucrați doar texte și imagini nu foarte complicate, sistemul Windows98 este preferabil. Sistemul WindowsXP, cu distincțiile lui între utilizatori obișnuiți și administrator, mai mult v-ar încurca.

A.1.1.1 Ce se găsește în ghidul introductiv Windows

Înainte de a ajunge la fișiere trebuie să ne familiarizăm cu sistemul Windows. Presupun că acesta a fost instalat de către firma de la care ați cumpărat PC-ul. De asemenea, firma v-a dat un CD autentic cu sistemul și o carte care vă arată pe scurt cum să vă descurcați cu sistemul. Trebuie să știți că fără sistemul de operare computerul este un simplu obiect decorativ sau o achiziție menită să vă facă să creșteți în ochii prietenilor.

Dacă n-ați lucrat niciodată cu un computer, urmați sfatul din ghidul introductiv Windows98 și studiați anexa despre utilizarea mouse-ului și capitolul al 3-lea, cel despre suprafața de lucru⁵.

În capitolul al 3-lea, fiți siguri c-ați înțeles exact rolul butonului Start și al pictogramelor My Computer și Network Neighborhood. Utilizarea lui Start este descrisă pe larg.

³În Tanenbaum și Woodhull, *op. cit.* găsiți în anexă 27646 de rânduri esențiale din sursa sistemului de operare Minix. Ca și-n anexele de față, fiecare rând de cod este numerotat. Puteți face astfel o comparație între micile fragmente de cod reproduse aici și dimensiunile programului care servește drept sistem de operare.

⁴Se pare că 95% dintre PC-urile din lume funcționează cu ajutorul Windows.

⁵În englezește, *desktop*.

apelăm. Sistemul Minix este descris de Tanenbaum și Woodhull.³ Prin contrast, sursele Windows sunt secrete. Firma care le realizează își protejează astfel drepturile sale de proprietate.

Dacă vrem doar să utilizăm un sistem de operare, atunci alte criterii intră în joc. Achiziția unei licențe pentru un sistem Windows98 sau pentru o versiune de casă a sistemului WindowsXP nu este chiar așa de costisitoare și costurile totale ale deținerii unui astfel de sistem sunt probabil sub cele ale unui sistem cu surse deschise. Multe periferice, precum scannerile sau aparatele foto, se integrează foarte ușor sub Windows. Nu se poate spune același lucru despre Linux.

Cum cea mai mare parte a publicului care consultă acest ghid lucrează cu un sistem Windows, soluțiile descrise aici sunt cele valabile pe acest sistem.⁴ Presupunerea noastră a fost una minimală. Am pornit de la ideea că sistemul utilizat este Windows98. De altfel, dacă nu sunteți conectați direct la Internet și prelucrați doar texte și imagini nu foarte complicate, sistemul Windows98 este preferabil. Sistemul WindowsXP, cu distincțiile lui între utilizatori obișnuiți și administrator, mai mult v-ar încurca.

A.1.1.1 Ce se găsește în ghidul introductiv Windows

Înainte de a ajunge la fișiere trebuie să ne familiarizăm cu sistemul Windows. Presupun că acesta a fost instalat de către firma de la care ați cumpărat PC-ul. De asemenea, firma v-a dat un CD autentic cu sistemul și o carte care vă arată pe scurt cum să vă descurcați cu sistemul. Trebuie să știți că fără sistemul de operare computerul este un simplu obiect decorativ sau o achiziție menită să vă facă să creșteți în ochii prietenilor.

Dacă n-ați lucrat niciodată cu un computer, urmați sfatul din ghidul introductiv Windows98 și studiați anexa despre utilizarea mouse-ului și capitolul al 3-lea, cel despre suprafața de lucru⁵.

În capitolul al 3-lea, fiți siguri că înțelegeți exact rolul butonului Start și al pictogramelor My Computer și Network Neighborhood. Utilizarea lui Start este descrisă pe larg.

³În Tanenbaum și Woodhull, *op. cit.* găsiți în anexă 27646 de rânduri esențiale din sursa sistemului de operare Minix. Ca și-n anexele de față, fiecare rând de cod este numerotat. Puteți face astfel o comparație între micile fragmente de cod reproduse aici și dimensiunile programului care servește drept sistem de operare.

⁴Se pare că 95% dintre PC-urile din lume funcționează cu ajutorul Windows.

⁵În englezește, *desktop*.

Esențial este apoi să înțelegeți felul în care se creează, se caută și se deschid dosarele cu fișiere și fișierele.⁶

Ghidul introductiv Windows98 vă explică, de asemenea, pe scurt ce anume este Internet-ul. Vom presupune, de asemenea, că utilizați în mod curent Internet Explorer 6.⁷

A.1.1.1.1 Taste funcționale și combinații de taste în Windows Unul dintre avantajele interfețelor grafice este faptul că rolul meniurilor și butoanelor este explicat chiar de numele lor sau de pictogramele asociate. De multe ori este însă mai ușor de lucrat cu taste și combinații de taste.

Tastele cu utilizare generală le găsiți listate în finalul ghidului Windows98. Dată fiind importanța lor, vom descrie și noi câteva taste.

F10 servește la activarea meniurilor. Observați, de asemenea, că
15 – în meniuri – literele subliniate indică tasta pe care puteți apăsa pentru a lansa o comandă. **meniuri**

CTRL+C servește la copierea unei porțiuni selectate dintr-un fișier. Fiți sigure c-ați înțeles ce înseamnă selecție înainte de a utiliza combinațiile care urmează. **copiere**

20 CTRL+X servește la decuparea unei porțiuni selectate. Nu trebuie să vă inducă în eroare dispariția de pe ecran a porțiunii selectate. Ea este disponibilă pentru inserare în alt punct, în momentul următor. **decupare**

CTRL+V servește la inserarea unei porțiuni copiate sau decupate, în punctul în care se află cursorul. **inserare**

CTRL+Z servește la anularea acțiunii pe care ați întreprins-o în momentul anterior. Este extrem de util să știți de existența ei, din motive lesne de înțeles. **anulare**

30 Acestea sunt combinații de taste valabile în general în programele Windows (în ferestrele pe care le deschide Windows).

Ca să vă mișcați între ferestre vă puteți sluji de ALT+TAB. Evident, clicurile cu mouse-ul vă permit să faceți același lucru.

35 Mouse-ul nu este întotdeauna o unealtă cu care poți lucra precis. Comparați selecția cu ajutorul mouse-ului și selecția pe care o puteți face ținând SHIFT apăsat și lucrând cu săgețile sau PG UP, PG DN.

⁶Termenul englezesc pentru „dosar“ este *folder*. Cel pentru „fișier“ este *file*.

⁷Dacă versiunea preinstalată este 5, puteți aduce lesne Internet Explorer la o versiune superioară.

A.1.1.1.2 Taste speciale Windows Dacă aveți o tastatură specială Windows, atunci găsiți pe rândul de jos al tastaturii două taste speciale Windows.

Tasta din stânga, cea cu steagul Windows pe ea, dacă este apăsată, derulează meniul de start.

meniul
con-
textual

Tasta din partea dreaptă, cea cu o foaie și săgeată pe ea, este poate mai rar observată, dar poate fi extrem de utilă. Selectați pictograma My Computer de pe suprafața de lucru. Apăsați tasta respectivă. Apare un meniu.

Același meniu apare și la un clic pe butonul dreapta al mouse-ului. Acest meniu se numește „meniu contextual“.

Meniurile contextuale sunt deosebit de importante și fără a fi familiarizată sau familiarizat cu ele practic nu poți lucra eficient în Windows. De ce? Un fișier poate fi deschis de către mai multe programe. Ai nevoie de meniul contextual pentru a avea acces rapid la programele care prelucrează fișierul respectiv. De asemenea, meniul contextual vă oferă acces la proprietățile obiectelor cu care sistemul de operare a populat computerul dumneavoastră.

A.1.1.1.3 Amenajarea suprafeței de lucru Ghidul primit o dată cu achiziționarea Windows98 vă oferă unele sugestii referitoare la amenajarea suprafeței de lucru.

imaginea
de pe
suprafața
de lucru

Cu puțin noroc puteți înfrumuseța lesne suprafața de lucru. Căutați pe Internet un situl <<http://www.webshots.com>>. Descărcați de acolo una sau mai multe dintre colecțiile de imagini pentru suprafața de lucru. Instrucțiunile de instalare le găsiți pe sit. Puteți adăuga și propriile imagini.

A.1.1.2 Partiționarea discului dur și segmentarea sistemului

S-ar putea ca multora această secțiune să li se pară vrăjitoarească. În realitate, cred că este bine ca să vă gândiți la cele scrise aici încă din prima fază a amenajării sistemului dumneavoastră de calcul.

Aș putea adăuga că vă puteți gândi la aceste lucruri chiar atunci când achiziționați componentele sistemului. De ce? Pentru că puteți primi nu doar CD-ul cu sistemul Windows, ci și multe alte programe utile care sunt, cum se spune în engleză, *bundled* (atașate) unor componente. De pildă, o placă de baza poate avea atașat un program de partiționare a discului dur. Programele acestea atașate le

puteți folosi în condiții perfect legale doar împreună cu echipamentul procurat.

Vă întrebați probabil la ce v-ar folosi partiționarea discului dur? Ei bine, dacă aveți un singur disc dur, programul de partiționare va crea mai multe discuri virtuale. Este ca și cum ați avea mai multe discuri în computer.

Pot surveni accidente care s-ar putea să vă determine să ștergeți sistemul Windows și să-l reinstalați de la zero. În asemenea condiții este mult mai sigur să aveți sistemul de operare pe un disc separat. Citiți însă și recitiți instrucțiunile programului de partiționare. Puteți nu doar să distrugeți date prețioase, dar chiar să faceți ceva care ar duce la nerecunoașterea discului dur de către sistem.

Avariile teribile sînt mai mult o chestiune de principiu. Practica de zi cu zi ne furnizează alte motive pentru a partiționa discul dur. Putem avea, de exemplu, trei discuri virtuale. Discul **c** este dedicat sistemului Windows și programelor care sunt atașate diverselor componente. Pe discul **d** sunt puse programele cu sursă deschisă.⁸ De asemenea, aici sunt adăpostite datele importante. Discul **e** este folosit pentru stocarea temporară de date și testarea de programe.

Discurile **d** și **e** trebuie să aibă însă și un sistem de dosare judicios organizat. Dosarele create de mine au nume scurte. De regulă, două-trei litere sugestive sunt ușor de ținut minte și este ușor de tastat ceea ce numește tehnic „calea” către un fișier (în engleză termenul tehnic este **path**). Un exemplu de genul **D:\use\bin** ne arată imediat cum desemnează Windows căile: mai întâi este indicat discul, apoi se pune bara oblică inversă și numele unui dosar și așa mai departe.

Folosirea \ în numele de căi este exact opusă celei din sistemele Unix, unde se utilizează în același scop /. Încurcăturile create sunt numeroase, dar trebuie să învățăm să ne descurcăm. De asemenea, sistemele Unix nu folosesc conceptul de disc. Totul este organizat pe un singur arbore, a cărui rădăcină este notată cu /. Dosarele sunt organizate foarte disciplinat. Dosarele **bin**, de pildă, sunt strict pentru programe executabile.

Pentru o bună compatibilitate cu programele din lumea Unix nu dați, de asemenea, nume de dosare cu spații. Folosiți spațiul subliniat _ acolo unde vreți să introduceți un spațiu. Un nume de dosar

partițiile

calea

**căi în
Unix**

**nu dați
dosarelor
nume cu
spații**

⁸Pentru ideea de sursă deschisă v. §A.1.5.

care are spații, dacă apare în calea invocată de către un program, va crea confuzii în sistem.⁹

În rezumat, reorganizarea și segmentarea sistemului au și scopul de pregăti condițiile pentru un bun transfer al programelor din lumea Unix. Aveți astfel la dispoziție și sistemul Windows, cu posibilitățile lui de a manevra sofisticat interfețele grafice, și elemente ale sistemului Unix, mai orientat către programare. Calculatorul se transformă într-o veritabilă stație de lucru.

adăpostiți **A.1.1.2.1 Copiile de siguranță** Dacă aveți un fișier sau un
periodic folder important, atunci este bine să-l salvați periodic într-un loc
fișierele sigur. Din acest punct de vedere, partiționarea discului dur joacă un
prețioase rol important. Chiar dacă ar trebui să reinstalați de la zero sistemul
de operare, aveți un spațiu pe care să adăpostiți fișierele prețioase.

Inscripționarea de CD-uri este o altă soluție. Un CD poate adăposti o mare cantitate de date.

splitfile Dacă nu dispuneți de un inscriptor de CD, atunci un program care poate să împartă un fișier mare în bucăți care încap pe o dischetă este foarte util. Veți găsi cu siguranță că programul **Splitfile** al lui Magnus Nilsson¹⁰ joacă bine acest rol.

Splitfile ocupă un loc foarte mic pe o dischetă. Nu depindeți de nici un fel de program instalat pe un computer sau altul.¹¹ Singura precauție de care trebuie să țineți cont este aceea de a nu încerca să umpleți la maximum o dischetă. Este mai prudent să indicați o valoare mai mică de 1.44MB pentru bucățile de fișier.

O alternativă, pe care am testat-o mai puțin, dar care pare o soluție excelentă, este programul **SplitFile** creat de către Jitendra Garodia.¹² Programul are mai multe posibilități decât cel al lui Nilsson. De asemenea, formatul fișierelor create este compatibil cu cel

⁹Puteti păți acest lucru și când treceți un fișier din Windows98 în Windows2000, nu doar când apelați programe Unix sub Windows.

¹⁰Programul Splitfile nu pare să aibă o pagină de web proprie. Ultima dată (12/01/2004) l-am găsit la adresa <<http://biphome.spray.se/mason40/index.html>>. Trebuie căutat și cu ajutorul cuvintelor-cheie.

¹¹Versiunea 1.4.1.12 a 2xExplorer dispune și ea de un **Split file** intern. Nu uitați însă că vă trebuie 2xExplorer și pe computerul pe care vreți să refaceți fișierul decupat în bucăți! De asemenea, formatul folosit de 2xExplorer nu este compatibil cu cel al lui Nilsson.

¹²Pagina sa de web este la <<http://jituonline.freesevers.com>>.

din 2xExplorer.¹³ SplitFile creează o arhivă desfăcută în mai multe fișiere; cu această arhivă poate lucra și 7zip.¹⁴

În rezumat, morala principală este foarte simplă: nu lucrați fără a face, în prealabil, o copie a fișierului prelucrat, fie pe alt disc dur, fie pe alt calculator, fie pe un CD, fie pe dischete. Puneți periodic la adăpost fișierele cu date prețioase. În orice caz, salvați din când în când fișierele pe discul dur¹⁵

A.1.2 De la fișele tradiționale la fișierele computere- lor

10 Munca intelectuală tradițională stă sub semnul munților de fișe. Era nevoie de zeci de sertare, de tot felul de cutii pentru a stoca fișele. Aveai nevoie de fișe pentru a-ți consemna ideile proprii, pentru a sistematiza ideile altora, pentru a crea liste bibliografice, note și pentru a indexa cărți. Doar manuscrisul ca atare era așternut pe coli
15 de scris de format A4.

Deși este doar o analogie, asemuirea unui fișier cu o colecție de fișe este cât se poate de utilă. Ea ne ajută să facem legătura între munca intelectuală de factură tradițională și cea efectuată cu ajutorul calculatorului.

20 Fișierele cele mai importante pentru noi vor fi cele zise „fișiere plate”. În esență, acestea sunt fișiere de tip text, în care sunt introduse eventual anumite marcaje structurale, dar care pot fi citite aidoma unor texte.

Dacă vă gândiți la fișele pe care vă notați ideile, atunci puteți face
25 o analogie între ele și rândurile dintr-un fișier de tip text. Analogia aceasta nu este lipsită de ambiguități,¹⁶ dar este utilă.

Ca lucrurile să fie mai încurcate decât par, sfârșitul de rând în sens logic nu este marcat la fel pe cele trei platforme importante (Windows, Unix și Mac). Într-un fișier de calculator totul are
30 un cod. Dacă am crea un fișier `woo.txt` în Windows sau în predecesorul acestuia, MS-DOS, am vedea că sfârșitul de rând este marcat

**fișierele
electro-
nice ca o
colecție
de fișe
tradițio-
nale**

CR LF

¹³ Atenție doar la faptul că arhivele generate de 2xExplorer trebuie redenumite. Extensia `.1` trebuie transformată în `.001` și așa mai departe, conform extensiilor produse de SplitFile sau 7zip.

¹⁴ Arhiva este compatibilă cu cea generată de către `split.exe` din setul de unelte Unix, în versiunea GnuWin32. Acesta este un program cu sursă deschisă, foarte cunoscut și bine testat. Este mai greu de folosit însă de către începători.

¹⁵ Căderile de curent pot fi inamicul cel mai periculos. Dacă mâine trebuie să predați eseu și sunteți aproape gata, o cădere de curent poate să compromită totul, în cazul când n-ați fost prudentă sau prudent.

¹⁶ V. distincția dintre rânduri logice și rânduri vizuale în §A.2.1.2.

LF prin două coduri. Numele convențional al acestora este CR LF. Un
fișier analog, `uoo.txt`, creat sub Unix, are marcat sfârșitul de rând
logic cu LF¹⁷. Iar fișierul analog, `moo.txt`, creat sub Mac, are marcat
CR sfârșitul de rând logic cu CR¹⁸.

A.1.2.1 Managementul fișierelor

5

Mai importantă decât înfrumusețarea suprafeței de lucru este extinderea posibilităților de a explora discul dur al computerului. Sub Windows98 și WindowsXP există un program special destinat explorării fișierelor și pe care-l găsim în meniul contextual atașat lui My Computer dacă apăsăm pe Open sau Explore.

10

**extensiile
fișierelor** Programul de explorare a fișierelor este setat automat la instalare după o formulă cam paternalistă. Cea mai utilă schimbare mi se pare a fi afișarea extensiei tuturor numelor de fișiere. Windows folosește extensia (literele puse după un punct în numele fișierului pentru a determina tipul acestuia și acțiunile asociate). Ca să vedem toate extensiile trebuie să parcurgem ruta Start → Settings → Folder Options. . . În Folder Options apăsăm pe View și deselectăm Hide file extensions for known file types.

15

WindowsXP include Folder Options. . . în Control Panel.

Ucenicii vrăjitori nu vor rezista desigur tentației de a vedea fișierele ascunse ale sistemului și vor selecta opțiunea Show all files. Atenție însă la modificările pe care le faceți în fișierel^é care vă sunt dezvăluite prin această manevră.

20

A.1.2.1.1 2xExplorer Programul de explorare inclus în Windows este oarecum limitat. Ar trebuie să aveți un veritabil manager de fișiere.

25

Programul care mie mi se pare cel mai potrivit pentru acest scop este 2xExplorer. Este scris de Nikos Bozinis.¹⁹ Instalarea programului lui Bozinis este extrem de simplă,²⁰ dar în faza în care presupun că vă aflați acum s-ar putea să vă vină mai greu să faceți acest lucru sub Windows98.²¹ WindowsXP dispune de un program de dezarhi-

30

¹⁷De la expresia englezească *line feed*.

¹⁸De la expresia englezească *carriage return*, care avea un sens evident pe vremea mașinilor de scris.

¹⁹Îl puteți găsi la adresa <<http://www.netez.com/2xExplorer>>.

²⁰Versiunea 1.4.1.12 dispune și de un instalator. Eu am folosit însă, în continuare, dezarhivarea, copierea fișierelor și generarea de scurtături.

²¹Citiți secțiunea A.1.2.2 dacă nu știți să lucrați cu arhive (cu fișiere compri-mate).

vare așa că ajunge să indicați un dosar în care să fie despachetate fișierele programului. Ca să-l porniți comod puteți crea o scurtătură pe suprafața de lucru.

De îndată ce ați făcut dublu clic pe scurtătura nou creată apare o fereastră divizată în trei panouri. Eu am închis însă panoul din stânga apăsând semnul × din colțul din dreapta sus. Am tras, de asemenea, bara cu unelte în partea de sus. Oricum, aceste lucruri sunt mai puțin importante în sine. Fiecare poate aranja cum vrea aspectul ferestrei în care apare 2xExplorer.

La ce este bun 2xExplorer sau un program similar? Ușurează enorm managementul dosarelor și fișierelor. Am văzut mulți utilizatori care pun totul pe suprafața de lucru, în dosarul *My Documents*, creat automat la instalarea Windows, sau care plasează fișierele direct în rădăcina discului c. Aceste obiceiuri denotă faptul că nu au făcut un efort pentru a înțelege rolul managementului fișierelor. În orice caz, nici să nu vă gândiți să folosiți uneltele transferate din Unix fără managementul adecvat al fișierelor. Riscați să produceți sute de fișiere în același dosar, printre care vă va fi absolut imposibil să vă descurcați.

20 A.1.2.1.1.1 Crearea de dosare și fișiere cu 2xExplorer

Pentru început deschideți 2xExplorer și verificați dacă înțelegeți cum puteți crea dosare și fișiere. Navigați către un loc care nu este periculos.²² Apăsați F8 și veți vedea cum apare un dosar pe care scrie *New Folder*. Schimbați-i numele în „Test”.²³ În 2xExplorer exersați acum clicul simplu (apăsarea) pe butonul din stânga al mouse-ului, cu săgeata îndreptată către un dosar. Veți vedea cum se schimbă culoarea pictogramei dosarului. Dacă apăsați ENTER pătrundeți în dosarul respectiv. Cu BACK SPACE reveniți.

F8

Intrați acum în dosarul nou creat. Creați alte câteva dosare.

30 Intrați într-unul dintre aceste dosare. Apăsați F7.²⁴ Apare o pictogramă lângă care scrie *New File.txt*. Schimbați numele ca și-n cazul dosarului.

F7

Ce ați creat? Un fișier zis „de tip text”. Dacă dați un dublu clic pe un fișier de tip text și aveți o instalație Windows nemodificată, atunci se deschide fereastra unei aplicații numită Notepad. În tra-

Notepad

²²Altundeva decât în dosarul Windows.

²³Evident, puteți folosi oricare nume care vă convine. Recomandarea ar fi totuși să nu utilizați nume cu spații. De pildă, în loc de „New Folder” folosiți „New_Folder”. V. și pagina 125, rândul 36.

²⁴Versiunea mai nouă a 2xExplorer include și un meniu special *Actions*. Una dintre acțiuni este chiar crearea de fișiere.

ducere, numele programului înseamnă *carnetul de notițe*. Este chiar ceea ce sugerează pictograma acestui program.

Este bine să învățați folosirea Notepad explorând pe meniurile sale. Presupun că ați deprins folosirea în special a comenzilor *Copy*, *Cut*, *Paste*. Acest lucru este desigur imposibil fără să știți să selectați porțiuni dintr-un text. Exersați, de asemenea, copierea de text între diverse instanțe ale aplicației Notepad. 5

A.1.2.1.1.2 Operații cu fișiere Fișierele pot fi copiate (sub alt nume, chiar și-n același dosar!), pot fi mutate sau șterse. Aceste operații ar trebui să fie evidente pentru oricine s-a familiarizat cu Windows. 2xExplorer oferă o bară cu pictograme speciale pe care puteți da un clic atunci când vreți să executați astfel de operații. Funcționează și meniurile contextuale Windows. 10

A.1.2.1.1.3 Filtre Creați un dosar. Creați în el mai multe fișiere, unele cu extensia *txt*, altele cu *tex* (nu uitați punctul dinaintea extensiei!). 15

Mergeți acum, în 2xExplorer pe ruta *Files* → *Filters*. . . Apăsați pe tasta cu săgeata către dreapta. Puneți un punct și adăugați extensia *tex*. Apăsați ENTER. Sunt afișate acum numai fișierele de tip *tex*. 20

F12 A.1.2.1.1.4 Proprietățile fișierelor și dosarelor În 2x-Explorer selectați un fișier sau un dosar.²⁵ Apăsați tasta F12! Apare o fereastră cu proprietățile fișierului sau dosarului.

Uitați-vă mai ales la *Attributes*. De multe ori vrem să bifăm sau să debifăm *Read-only*. După cum sugerează și numele englezesc al acestei proprietăți, fișierul *read-only* poate fi doar citit. Dacă vrem să-l schimbăm, trebuie să eliminăm această proprietate. 25

A.1.2.1.1.5 Selectarea unui grup de fișiere O operațiune asemănătoare cu filtrarea este selectarea unui grup de fișiere. Mergeți în meniul principal la *Mark*. Alegeți *Select Group*. . . 30

tipare simple Acum trebuie să învățați doar să construiți tipare foarte simple. Semnul * ține locul unui grup de litere. Semnul ? ține locul unei sin-

²⁵ Studiați felul în care sunt setate clicurile pe sistemul dumneavoastră. Un clic pe pictograma din stânga numelui este mai prudent. Clicul pe nume sau un dublu clic rapid pe nume s-ar putea să deschidă o mică fereastră pentru operarea de schimbări în nume.

gure litere. Dacă scrieți *.tex, atunci vor fi selectate toate fișierele de tip tex.

A.1.2.1.1.6 Informații despre un dosar cu fișiere În meniul principal dați clic pe Tools și apoi pe Folder Data. . . . Veți obține date despre dosarul respectiv.

La ce sunt bune aceste date? Cea mai simplă utilizare este legată de folosirea spațiului de pe discul dur. De pildă, pe computerul pe care scriu, dosarul cu complexul de programe T_EX ocupă 653.5MB. Pot însă vedea și arborele subdosarelor din dosarul respectiv, cu statistica numărului de dosare și de fișiere din ele.

Datele astfel obținute pot fi păstrate într-un fișier de tip text. Pentru aceasta folosiți meniul contextual al ferestrei cu date despre dosare. Dați un clic pe Print. În ciuda numelui, aveți posibilitatea să creați un fișier. Puteți face acest lucru și pentru un CD-ROM.

Păstrați documentele descărcate de pe Internet în dosare care au ca nume data descărcării. În aceste dosare grupați fișierele în dosare cu nume semnificative. Adăugați, dacă este necesar, într-un fișier special adresa de Internet și date cu privire la locul de unde ați descărcat documentele. Creați cataloage cu aceste dosare după metoda descrisă mai sus. În acest fel nu veți avea probleme cu citarea documentelor de pe Internet.

A.1.2.1.1.7 Căutarea de fișiere O operație pe care o întreprindem adesea este căutarea de fișiere. Mergeți la Tools în 2xExplorer. Căutați un fișier al cărui nume nu-l știți. Puneți doar *.* , semn că nu știți nici numele, nici extensia. Știți însă un fragment de text din fișier. Introduceți fragmentul de text respectiv în caseta de dialog, după ce ați bifat **use text constraints**.

Căutarea pe care o oferă 2xExplorer nu este mult mai sofisticată decât cea pe care ați găsi-o la **Start**→**F**ind în Windows, dar o puteți efectua mai ușor, direct în dosarul care vă interesează.

A.1.2.1.1.8 Aranjarea meniului de start Activați panoul din stânga al 2xExplorer-ului. Urmați ruta **Bookmarks**→**G**o to folder→**S**tart Menu și dați clic pe **S**tart Menu. Creați acum dosare în care grupați scurtături către programele pe care le-ați instalat.

Puteți crea noi scurtături într-un mod cât se poate de simplu. În panoul din dreapta mergeți la un dosar unde aveți un executabil Windows. Selectați-l! Mergeți la **E**dit și copiați (CTRL+C).

Treceți apoi în panoul din stânga. Mergeți la Edit și apăsați Paste Shortcut. Acum puteți redenumi scurtătura.

Nu uitați sfatul de a folosi un dosar și programe destinate doar testării până sunteți siguri și siguri pe ceea ce faceți.

A.1.2.2 Managementul Fișierelor Comprimate

5

Chiar dacă n-aveți o idee prea exactă despre fișiere vă puteți da seama că ele ocupă un loc în mediile de stocare. Arhivarea este un proces prin care fișierele sunt comprimate. Mai mult decât atât, o colecție de fișiere este strânsă la un loc și circulă sub forma unui pachet unic.

10

WinXP dispune de un sistem de dezarhivare propriu. Lucrul acesta nu este însă adevărat pentru mai vechiul Win98. Pe Internet sau pe CD-urile atașate revistelor dedicate lumii calculatoarelor veți găsi lesne destul de multe variante gratuite de programe de arhivare.

7-zip

O soluție care are numeroase avantaje o reprezintă programul 7-zip, creat de Igor Pavlov. Pagina de web a acestui program este <<http://www.7-zip.org>>. Consultați, de asemenea, <<http://sf.net>>.

15

7-zip are propriul manager de fișiere. Cu ajutorul acestui manager puteți căuta în interiorul unui pachet de fișiere.

20

De asemenea, vă puteți sluji de comenzile pe care programul le instalează în meniul atașat oricărui fișier (meniul care apare când faceți clic pe dreapta pe pictograma fișierului).

O problemă este lipsa unui dezarhivator pentru arhivele specifice Mac. O soluție este Aladdin Expander al firmei AladdinSystems. În acest caz, citiți cu atenție licența produsului. Acest lucru este valabil, evident, pentru toate programele pe care le vom menționa aici. În special fiți atente și atenți la faptul că n-aveți dreptul să obțineți un profit comercial în urma distribuiri produsului.²⁶

25

A.1.3 Vizualizarea fișierelor

30

Probabil v-ați familiarizat deja cu sistemul de extensii al fișierelor în Windows. Extensia este grupul de litere care urmează după punctul pus la sfârșitul numelui fișierului.

²⁶Cu alte cuvinte, dacă ați descărcat de pe Internet acest dezarhivator sau l-ați obținut în alt mod nu-l puteți vinde altcuiva. Evident, puteți să-l dați mai departe, dar nu puteți cere o compensație financiară decât pentru mediul de stocare sau pentru alte cheltuieli implicate de transferul produsului.

În 2xExplorer puteți vizualiza un fișier dacă-l selectați și apăsați apoi tasta F3. Evident, acesta este un mod foarte limitat de a vizualiza fișiere și vă va fi de folos doar pentru cele care conțin doar texte simple. Pentru alte fișiere aveți nevoie de programe de vizualizare dedicate unui anumit tip de fișier.

A.1.3.1 Vizualizarea fișierelor PostScript

Pe parcursul acestui ghid avem mare nevoie de fișiere cu extensia **ps**; denumirea completă a acestor fișiere este „fișiere PostScript“.

Pentru a vizualiza comod un fișier PostScript sub Windows aveți nevoie de două tipuri de programe. Primul este un interpretor al conținutului fișierului.²⁷ Al doilea este programul de vizualizare ca atare.

Descărcați de pe Internet unul dintre fișierele **gs700w32.exe**, **gs704w32.exe**, **gs800w32.exe** sau ceva mai nou din această serie. Citiți eventuale instrucțiuni de instalare. Oricum, procesul de instalare este extrem de simplu. Un fișier **exe** este executabil direct în Windows. După ce ați plasat convenabil fișierele rezultate mergeți la dosarul **doc** și citiți fișierul **PUBLIC**. El conține **Aladdin Free Public License**, pe scurt **AFPL**. Comparați-o pe loc sau ulterior cu importanta licență **GNU** a **Free Software Foundation**.

Pentru început n-are sens să încercați să faceți mare lucru cu programul pe care l-ați instalat. Este doar bine să știți un se află, care este versiunea sa și că se numește Ghostscript.

Pentru a instala vizualizatorul descărcați un fișier cu un nume de genul **gsv44w32.exe**. Îl găsiți în multe locuri pe Internet. Primul număr indică versiunea. Aveți grijă să fie compatibilă cu versiunea programului Ghostscript. Din nou, instalarea este relativ simplă. Citiți însă instrucțiunile. Va trebui să indicați însă ce versiune a Ghostscript vreți să folosiți. Programul pe care tocmai l-ați instalat se numește Ghostview. Este foarte folosit în lumea academică, dar are o licență, ca să zic așa, „la limită“ programelor care vor fi recomandate în această carte. Pentru a face să dispară o fereastră care tot apare când deschideți programul²⁸ trebuie să înregistrați programul contra unei sume de bani. Versiunea neînregistrată este însă absolut funcțională și nu cuprinde nici un fel de limitări.

Versiunile mai noi ale Ghostview nu integrează însă o unealtă ex- pstoedit

²⁷PostScript este un limbaj creat de firma Adobe pentru tipărirea de documente pe printere cu laser. Limbajul poate fi folosit și pentru crearea de imagini pe ecran. Pentru o scurtă prezentare a limbajului PostScript v. §C.1.2.

²⁸Acestui tip de fereastră i se spune în engleză *nagscreen*.

trem de necesară denumită „pstoedit“. Despre pstoedit puteți citi la <http://www.pstoedit.net/pstoedit/>. Concret, pentru instalare, vă trebuie fișierul **pstoeditsetup.exe** și, desigur, priceperea minimă necesară pentru a instala programe sub Win98.²⁹

Ce face pstoedit? Vă oferă posibilitatea de a converti un fișier PostScript într-unul de tip pdf. 5

A.1.3.2 Vizualizarea fișierelor pdf

Pdf este o prescurtare a unui termen englezesc care ne spune că este vorba despre fișiere care stochează documente într-un format portabil. Portabilitatea înseamnă, între altele, că aceste fișiere pot fi văzute atât sub Windows, cât și sub Unix și Macintosh, cele trei platforme principale în lumea calculatoarelor. 10

Acrobat Reader Tot atât de importantă ca și portabilitatea este calitatea fișierelor pdf de a fi veritabile cărți electronice. Pentru a vedea aceste cărți trebuie să instalați un sistem de vizualizare. Cel mai faimos este Adobe Acrobat Reader. Citiți despre acest program la adresa <http://www.adobe.com/products/acrobat>. Ca mai toate programele de vizualizare, Acrobat Reader este distribuit gratuit. 15

Pentru a avea pe deplin senzația cărții electronice puteți instala Adobe Acrobat eBook Reader. 20

A.1.3.3 Vizualizarea altor tipuri de fișiere

Programele de vizualizare a fișierelor, spre deosebire de cele de editare, sunt de regulă oferite gratuit de către firme.

Word Viewer În mod sigur veți avea nevoie periodic de un vizualizator de fișiere doc. Microsoft Word Viewer poate fi folosit în mod gratuit. Puteți vedea cu el și fișierele de tip rtf. 25

OpenOffice OpenOffice este o suită completă de aplicații pentru birou; poate fi folosită atât pentru vizualizarea, cât și pentru editarea de fișiere doc și rtf.³⁰

Microsoft Reader Pe Internet veți găsi, de asemenea, fișiere de tip lit. Pentru a le vedea aveți nevoie de Microsoft Reader.³¹ 30

²⁹Între altele, instalarea sub WinXP este mai complicată. Trebuie să studiați în mod special cum se pot instala programe sub asemenea sistem de operare.

³⁰Puteți să vă procurați OpenOffice de la adresa <http://www.openoffice.org>.

³¹Puteți descărca programul de instalare de pe situl firmei la <http://www.microsoft.com>.

Un alt tip de fișiere pe care s-ar putea să le întâlniți pe Internet sunt cele pentru **palm computere**. Există cărți întregi în format **pdb**. Un vizualizator gratuit pe care l-am folosit cu succes este DocReader-ul creat de Mike Pickering.³²

Doc-Reader

- 5 Evident, pentru omniprezentele, pe Internet, fișiere **html**, este nevoie de un vizualizator pentru asemenea fișiere. De regulă, Microsoft Internet Explorer sau Netscape Navigator sunt folosite pentru a vizualiza asemenea fișiere.

Internet Explorer și/sau Netscape Navigator

A.1.4 Cine se teme de utilizarea computerelor?

- 10 Pentru a adopta perspectiva programatorului trebuie să știm ceva mai mult despre utilizarea Windows.

- Mergeți la **Start** → **Run**. Tastați cuvântul **command** și apoi apăsați **ENTER**. Apare o fereastră neagră ca aceea din vechiul MS-DOS. În această fereastră nu pot fi date decât comenzile numite „comenzi în linie”. Dacă nu știți, tocmai ați dat o asemenea comandă pentru a chema fereastra MS-DOS.

command

- Reveniți la **Start** → **Run**. Apăsați butonul **Browse**. Navigați un pic până găsiți pe discul **C** dosarul **Windows**. Apoi căutați **NOTEPAD.EXE** și când îl găsiți dați un clic pentru selecție și apăsați butonul **Open**. Aranjați fereastra neagră a MS-DOS prompt-ului în așa fel încât să vedeți mica fereastră **Run** deschisă în colțul din stânga-jos al ecranului. Mergeți în fereastra neagră a MS-DOS. Tastați cu grijă comanda din fereastra **Run**. Verificați dacă înțelegeți conceptul de cursor³³ Puteți recurge și la procedura de copiere și inserare de text dintr-o fereastră într-alta. După ce ați scris sau copiat textul relativ misterios, apăsați tasta **ENTER**. Se va deschide o fereastră **Notepad**.³⁴

exemplu de comandă în linie

- Ce mai putem scrie în fereastra neagră? Tastați **echo Un text**. Programul va afișa cuminte **Un text**. Cuvântul **echo** este o comandă, care spune programului să afișeze textul care urmează imediat după comanda respectivă.

istoria comenzilor

Putem influența ceea ce se întâmplă la prompt-ul MS-DOS? Da. Merită să dispunem de o istorie a comenzilor. Mergeți pe discul **C**

³²Îl puteți descărca de la <http://www.crosswinds.net/~mpicker0>.

³³O liniuță clipește în fereastra neagră atunci când aceasta este fereastra în care lucrați. Această liniuță de subliniere indică punctul în care se află cursorul. Este punctul în care, practic, puteți insera un text.

³⁴Am făcut același test și-n WindowsXP. Nu au fost probleme. Copierea în fereastra neagră diferă desigur. Practic trebuie recurs la meniul obținut prin clic pe butonul din dreapta al mouse-ului.

chiar în rădăcină și plasați săgeata cursorului Windows pe `AUTOEXEC.BAT`. Faceți clic pe dreapta. Faceți apoi clic pe Edit. Fișierul acesta este un fișier de tip text, dar al cărui conținut este, pentru sistemul de operare, un șir de comenzi. Adăugați la sfârșitul fișierului comanda `c:\windows\command\doskey.com`. Verificați dacă efectiv aveți pe calea indicată un fișier cu numele `doskey.com`. Reporniți calculatorul.³⁵ 5

Când sistemul a repornit deschideți din nou o fereastră MS-DOS. Dați ce comenzi vreți în ea. Apăsăți apoi tasta cu o săgeată în sus și pe cea cu săgeata în jos. Observați că acum sistemul ține minte comenzile.³⁶ Cu tastele cu săgeți către stânga și către dreapta vă puteți mișca prin textul comenzii. Folosiți tasta `Ins`³⁷ pentru a modifica o comandă fără a o scrie de la capăt. 10

Ați dat deja peste un fel de resursă secretă și extrem de utilă a sistemului MS-DOS. 15

A.1.4.1 Comenzile în linie din 2xExplorer

În multe situații s-ar putea să aveți nevoie să dați rapid o comandă în linie. Putem folosi pentru a atinge acest scop și 2xExplorer-ul.

**Run
Command**

Cea mai simplă comandă în linie este formată din numele unui program executabil. Mergeți la **Tools** și dați clic pe **Run Command**. Tastați `sfc` și apoi `ENTER`. Apare unul dintre cele mai utile programe din Windows98. Acest program verifică integritatea sistemului. n-are nici un rost să reinstalați sistemul de câte ori aveți o pană. Îl puteți restaura cu `sfc`. 20

Dac-ați pus `splitfile` undeva în `PATH`, atunci puteți repeta comanda de mai sus cu `splitfile`. 25

Comenzile în linie de mai sus sunt utile pentru a chema programe Windows. Tastați însă `dir` urmat de apăsarea pe tasta `Enter`. Apare doar preț de o clipă o fereastră neagră. N-ați văzut nimic.

F10

`Dir` este o comandă în linie care trebuie dată într-o fereastră MS-DOS. Apăsăți tasta `F10` urmată de `Enter`. Apare o fereastră neagră (cel puțin aceasta este culoarea setată automat). Acum `dir` are alt efect. 30

Avantajul oferit de 2xExplorer poate din nou să pară minor.

³⁵ Windows vă va săcăi de multe ori cu aceste reporniri după instalări sau modificări în sistem.

³⁶ Le ține minte cât timp fereastra este deschisă.

³⁷ Tasta care alternează două moduri de scriere: unul prin inserarea de text; altul în care textul este suprascris.

Putem chema fereastra MS-DOS direct din sistemul Windows. Diferența este însă dată de faptul că ea are promptul chiar în dosarul în care am chemat-o. În plus, 2xExplorer are propria sa istorie a comenzilor. Avantajul nu este chiar mic.

5 A.1.4.2 Fișiere cu comenzi în 2xExplorer

O comandă în linie este un text ca oricare altul. De ce n-am putea crea un fișier cu asemenea comenzi?

Generate batch...

Mergeți în dosarul unde ați creat (cu titlu de probă!) fișiere cu extensia **tex**. Selectați-le! Mergeți în meniul principal la **Mark** și dați un clic pe **Generate batch...** Tastați comanda **ren \$N \$B.txt**. Dați clic pe **Create**. Salvați fișierul cu extensia **bat**. Dați comanda de executare a acestui fișier.

redenumirea fișierelor

Selectați fișierul **bat** creat și apăsați **F3**. Nu executați dublu clic pe un astfel de fișier pentru a-l deschide! Este un fișier cu comenzi! Se vor executa comenzile.

Nu este greu să înțelegeți structura comenzilor create. V-ar trebui un manual de MS-DOS pentru a crea fișiere **bat** mai complexe.³⁸ Oricum, redenumirea este una din operațiile de care te lovești foarte des. Este extrem de greu să o execuți manual.

O altă operație pe care uneori trebuie să o executăm asupra unui grup de fișiere este ștergerea. Comanda este **del**. Mare atenție însă la folosirea ei. Învățați pe fișiere și-n dosare cu fișiere create doar pentru teste. Testați în mod repetat ceea ce ați creat. Și nu blestemați autorul pentru efectele nedorite. Cartea nu este însoțită de nici un fel de garanții. Faceți totul pe propria răspundere.

ștergerea de fișiere

A.1.4.3 Programele ascunse din Windows

Există oare și resurse Windows mai mult sau mai puțin ascunse? Cum să nu! De unele este relativ periculos să vă apropiați. Altele sunt însă puțin cunoscute, dar extrem de utile.

Uneori există scurtături în **Accessories** pentru aplicații utile, dar ele nu sunt folosite prea des. Un exemplu este un program pe nume **kodakimg.exe**. Îl găsiți în dosarul Windows.

Puteți crea și propriile dosare cu scurtături în meniul **Start**. Mergeți în 2xExplorer pe următoarea rută cu meniuri: **Bookmarks**→ **Go to Folder**→ **Start Menu**. Creați un nou dosar și dați-i un nume su-

³⁸În WindowsXP există comenzile în linie specifice. Se pot obține explicații recurând la ajutorul oferit de sistem.

gestiv. Intrați în acest dosar și faceți o scurtătură către programul `kodaking.exe`.

tiff Când studiați meniul **Start** vedeți un meniu derulant în care găsiți scurtătura nou creată. Puteți porni acum un program de prelucrare de imagini. Acest program creează fișiere de tip **tiff**. Dacă aveți un scanner sau acces la un scanner și copiați electronic, să zicem, un manuscris de zece pagini, fișierul de tip **tiff** ține toate aceste pagini la un loc, într-un singur fișier. Puteți răsfoi acest fișier ca pe o carte. În plus, acesta este formatul folosit de către marea majoritate a programelor OCR, programele care recunosc literele. Aceste programe pot transforma un fișier de tip **tiff** într-unul de tip **text**.

sfc Un alt program deosebit de util este `sfc.exe`.³⁹ Îl găsiți tot în dosarul **Windows**.

MSCONFIG Alte programe ne duc limpede pe terenul unde se avântă doar ucenicii vrăjitori. Mergeți, de exemplu, din dosarul **Windows** în dosarul **System** și căutați acolo programul `MSCONFIG.EXE`. Acest program vă permite să configurați sistemul **Win98**. Evident, o faceți pe propriul risc.

SYSEDIT Cu `SYSEDIT.EXE` puteți edita fișierele de configurare a sistemului. Dacă nu sunteți ucenici vrăjitori, recomandarea ar fi să editați doar fișierul `AUTOEXEC.BAT` în modul indicat deja mai sus.

REGEDIT Un program pe care ucenicii vrăjitori îl vor încerca desigur este `REGEDIT.EXE`. Lansați acest program. Acum aveți acces la fișierele în care **Win98** își ține datele cu privire la programele instalate, tipuri de fișiere, moduri de deschidere a lor și așa mai departe. Nu umblați aici fără socoteală, deoarece aveți toate șansele să instalați haosul în sistemul dumneavoastră.

După ce ați lansat în execuție **Regedit**, dați clic pe cruciulița de la `HKEY_CLASSES_ROOT`. Folosiți glisorul pentru a ajunge la **Folder** și faceți iar clic pe cruciuliță. Puneți săgeata pe **Folder** și dați clic pe dreapta. Alegeți din meniu pe **New** ▸ **Key**. Numiți această cheie `2xExplorer`. Acum construiți o nouă cheie, dar cu rădăcina în `2xExplorer`, cheia pe care tocmai ați creat-o. Dați-i acestei chei numele `command`. Nu-i dați alt nume! Așa cum sugerează numele, aceasta este cheia pentru comanda în linie. Deschideți-o așa cum ați deschide orice dosar. Chemați un meniu prin clic pe dreapta valorii **Default**. Introduceți o comandă de tipul `D:\use\2xExplorer.exe`. Evident, puneți calea adecvată pentru sistemul dumneavoastră. Verificați eventual calea cu **Start** → **Run** → **Browse...**

Închideți **Regedit**. Obțineți un meniu prin clic pe dreapta pe

³⁹V. pagina 136, rândul 24

icoana cu numele calculatorului dumneavoastră. În meniu apare acum și 2xExplorer. Puteți renunța la pictograma de pe suprafața de lucru.

A.1.4.4 Strat Unix peste Windows

- 5 Înainte de a trece mai departe trebuie să ne reamintim un principiu important. Cartea aceasta vă îndeamnă să învățați cât mai mult experimentând cu diverse programe. Nu faceți acest lucru însă cu mai multe programe deodată. Instalați pe rând diverse programe și vedeți ce se întâmplă. Programele interacționează între ele și aceste
10 interacțiuni nu pot fi anticipate. Ați instalat poate în același dosar două fișiere care, dacă ar sta separat, rezultatul ar fi altul. Sau poate ați estimat eronat felul în care un program face apel la alt program. Nu vă grăbiți să puneți aceste efecte nedorite pe seama unor viruși.

programele interacționează

- De asemenea, dacă un program este nestabil, Win98 tinde să
15 se destabilizeze în totalitatea sa.⁴⁰ Nu este neapărat un motiv să renunțați la Win98 în favoarea unui sistem de operare care separă mai bine felul în care funcționează diversele programe. Win98 este rodul unui efort superb de a păstra compatibilitatea cu mai vechile programe MS-DOS. Poate, de asemenea, emula Unix în bune condiții.
20 Instalați foarte prudent noi programe.

Când ne apucăm să modificăm sistemul Windows98 este bine să știm să modificăm **AUTOEXEC.BAT**. Folosesc, din nou, exemplul computerului pe care este scrisă cartea: **Autoexec**

```
1 SET HOME=D:\dat\home
2 SET VIM=D:\USE\VIM
3 SET GHOSTSCRIPT_FONT_DIR=D:\ed\gs\fonts
4 SET PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;D:\USE\BIN
```

- Aceasta este o parte din fișierul **AUTOEXEC.BAT**. Citiți cu atenție
25 aceste rânduri, dar nu faceți modificări mecanice pe calculatorul dumneavoastră. Țineți cont de denumirile propriilor dosare.

HOME este numele unei variabile al cărei conținut este folosit de către diverse programe. Programele transferate din Unix folosesc dosarul indicat drept **HOME** pentru a plasa acolo fișiere de configurare.

- 30 Trebuie să fiți atente și atenți la modul în care se instalează Vim (și alte programe) în funcție de existența sau nu a unei valori pentru **HOME**. Editorul care este marea alternativă la Vim, Emacs, va cere în mod explicit să dați o valoare lui **HOME**.

⁴⁰ Avantajul WindowsXP constă, între altele, în eliminarea acestui tip de instabilitate.

Următorul punct important în setările de mai sus îl constituie SET PATH și căile care urmează după aceea (vedeți linia a patra din fragmentul din AUTOEXEC.BAT). Aceste setări vă scutesc de tastarea căii complete atunci când apălați un program printr-o comandă în linie. De exemplu, D:\USE\BIN va permite să apelez în linie orice executabil pus în dosarul BIN din dosarul USE aflat pe discul D. În dosarul respectiv pun, între altele, unelte aduse din lumea Unix.

Cum modificăm variabilele de mediu sub WindowsXP? În sistemul XP AUTOEXEC.BAT nu mai joacă rolul din MS-DOS sau Windows98. Mergeți pe ruta Start→ Control Panel→ System→ Advanced→ Environment Variables și ajungeți la o fereastră în care vedeți lista variabilelor de mediu. XP distinge între variabilele create de utilizatori și cele ale sistemului. De exemplu, eu am creat o variabilă HOME. Căutați pe PATH printre variabilele de sistem și selectați cu mouse-ul respectiva variabilă de mediu. Dați apoi un clic pe butonul Edit și adăugați calea pe care doriți să o știe sistemul.

A.1.4.4.1 Sistem Unix minimal sub Windows Resursele native Windows pentru comenzile în linie în fereastra MS-DOS sunt foarte limitate. O soluție o reprezintă instalarea de unelte Unix. Pentru început recomandabilă este instalarea unui strat minimal de unelte Unix.⁴¹

Puteți instala numai uneltele, fără compilatorul de C++. Dacă vreți să accesați direct uneltele Unix instalate, nu uitați să setați în mod corespunzător variabila PATH a sistemului.

Vom da o ilustrație foarte simplă a modului în care sunt folosite uneltele Unix. Vreți să știți, să zicem, câte cuvinte sunt într-un fișier. Apelați la programul wc.exe, al cărui nume vine de la expresia englezească word count. Mergeți, cu ajutorul 2xExplorer, în zona rezervată încercării programelor și creați un fișier test.txt în care scrieți câteva rânduri. Apăsați, în 2xExplorer, tasta F10 și scrieți comanda wc test.txt și apăsați tasta Enter. Puteți scrie, evident, comanda și direct în fereastra MS-DOS. Rezultatul constă dintr-o serie de cifre. Prima cifră indică numărul de rânduri; a doua numărul de cuvinte; a treia numărul de semne din fișier.

Vreți ceva ajutor? Scrieți comanda wc -help și studiați utilizarea opțiunilor. Parantezele drepte indică elementele comenzii care nu sunt obligatorii. Singurul mod de a învăța cu adevărat utilizarea

⁴¹Sistemul zis msys dispune de un program de instalare în stil Windows. Este excelent pentru început. Pentru detalii a se vedea <http://www.mingw.org/msys.shtml>.

unor astfel de programe este experimentatul. Numai așa vă puteți obișnui cu sistemul de opțiuni al comenzilor în linie.

A.1.4.4.2 GnuWin32 Sistemul `msys` nu folosește direct Windows. Există unelte Unix care folosesc în mod direct sistemul de
5 operare Windows. Proiectul GNU are o versiune pentru Windows. Găsiți în ea un set vast de unelte Unix, precum și multe alte programe.

Versiunea pentru Windows a proiectului GNU se numește „GnuWin32”. Consultați pagina de web cu pachetele de programe realizate în cadrul acestui proiect.⁴² Alegeți programele de care aveți
10 nevoie. Recomandarea noastră este ca, la început, să instalați doar programe care dispun de *setup*. Orice începătoare sau începător vor avea mari dificultăți la instalarea fără *setup*.

A.1.4.4.3 Programarea în C/C++ Ucenicii vrăjitori interesați de programarea în C++ vor fi desigur atrași de versiunea pentru
15 Windows a compilatorului `gcc`. Acest compilator este foarte important în sistemul de operare zis „Linux”. Este atât de important încât mulți cred că sistemului respectiv ar trebui să i se zică GNU/Linux. Numele celui care inițiat elaborarea `gcc` și mișcarea GNU, Richard
20 Stallman, ar trebui să fie cunoscut de toată lumea.⁴³

`gcc`

La ce este bun compilatorul? Nu vă convine modul cum se comportă `wc.exe`? Puteți rescrie programul.

Pe situl `<http://sourceforge.net>` găsiți un excelent mediu integrat de dezvoltare de programe în C/C++. Căutați un program
25 care se numește „Dev-C++”. Veți avea o interfață grafică tipică pentru Windows și compilatorul `gcc` integrat. Cu acest mediu puteți rescrie, evident după ce învățați C/C++, programele de genul celui de numărat cuvinte.

⁴² Adresa paginii de web cu lista de pachete GnuWin32 este `<http://gnuwin32.sourceforge.net/packages.html>`. Pentru surse și documentație, v.`<http://www.gnu.org>`, `<http://gnu.eunet.fi>` sau orice alt sit Internet legat de proiectul GNU.

⁴³ Pentru ideile mișcării GNU v. §A.1.5

A.1.5 Principiul surselor deschise

Sistemul de operare propriu-zis oferă puține posibilități de lucru.⁴⁴ Pentru a produce un eseu frumos tipărit sunt necesare o serie de alte programe.

Ce facem însă? Noi programe ar putea însemna noi cheltuieli. Dacă este vorba despre o firmă, atunci aceste cheltuieli ar putea să-și găsească justificarea: ele pot fi recuperate vânzând produsele firmei. Dacă este vorba despre o studentă sau un student, se profilează o dilemă foarte neplăcută. S-ar părea că alternativa este între a plăti sume considerabile și a pirata programe.

evitați Pirateria este însă hoție și ar trebui exclusă din start. Din păcate nu se întâmplă de multe ori așa. În ciuda accesului masiv la **pirateria** Internet,⁴⁵ sunt puțin folosite programele gratuite.⁴⁶

Gratuitatea nu este însă un scop în sine. Este foarte important să consultați licențele. Uneori produsul este gratuit doar pentru a-i face (un timp) reclamă. Versiunile ulterioare sau versiunile complete nu sunt gratuite. Recomandarea noastră, mai ales în cazul eseurilor studentești, este orientarea către o licență GNU sau o licență asemănătoare.

licența GNU este, în esență, o licență menită să asigure accesul la sursele programelor, să prevină plagiatul. Licența interzice, de asemenea, ascunderea rezultatelor făcute astfel publice într-un program care n-are surse deschise. Persoanele care au scris sursele sau părți ale acestora trebuie menționate în mod adecvat. De asemenea, cine operează o modificare în surse trebuie să arate acest lucru în mod explicit și să nu împiedice accesul altora la surse.

GNU nu este o licență menită să asigure gratuitatea programelor. Așa cum atunci când reproduci în editura proprie lucrarea altcuiva sau plătești pentru a citi o carte, nimic nu împiedică plata unei sume de bani pentru a avea acces la sursa unui program. În practică însă, multe dintre persoanele care au scris programe cu sursă deschisă au fost generoase și au plasat sursele și programele în domeniul public.

⁴⁴Linux este, de obicei, distribuit împreună cu o mulțime de programe de editare de texte, imagini etc. Cred că trebuie distins însă între aceste programe și sistemul de operare ca atare. Windows include, cu unele excepții, programe suplimentare foarte modeste și firma a fost chiar atacată în justiție din pricina integrării în distribuția de bază a Internet Explorer.

⁴⁵În Uniunea Europeană, în 2003, accesul la Internet, în diverse țări, variază între aproximativ 20% și 75%. La noi procentul este desigur mic, dacă numărăm casele și apartamentele conectate la Internet. Situația este diferită în cazul universităților.

⁴⁶Programele gratuite sunt distribuite, de asemenea, și de către reviste.

O serie de astfel de programe sunt rodul cercetării din mediul academic. Este absolut firesc ca, atunci când le cerem studentelor și studenților să tehnoredacteze un eseu, filosofic sau nu, să le arătăm și cum pot folosi aceste programe.⁴⁷

- 5 Pentru a înțelege mai bine sensul acestor principii ar trebui să clarificăm ce se înțelege prin *sursa* unui program.

Toți cei care sunt ahtiați după **forța** calculatorului lor discută despre **viteza** procesorului calculatorului. Mai importante sunt însă **minții** programele rulate pe calculatoare. Adevărata **forță** o reprezintă **mințile** care programează inteligent. Pentru programare sunt folosite **importante** limbaje de nivel mai înalt, care au anumite asemănări cu limbajele umane.

Sursele programelor sunt scrise în aceste limbaje de nivel mai înalt. Ele sunt stocate în fișiere de tip text.⁴⁸

- 15 În mediul academic și-n cercetare, în lipsa surselor, nu poate fi studiată decât utilizarea programelor. Lucrul acesta este absurd. Gândiți-vă la un program care, dacă îi dați două numere naturale, vă spune care este cel mai mare divizor comun al lor. N-aveți cum știți însă ce algoritm folosește. Este ca și cum am învăța matematica fără
20 a studia algoritmul lui Euclid. A studia doar utilizarea programelor ar echivala cu transformarea universităților în școli de meserii.

Există însă și alte avantaje ale surselor deschise în raport cu programele care sunt doar gratuite. Autoarea sau autorul unui program gratuit ar putea să înceteze să mai dezvolte produsul său. Aceasta
25 îi obligă pe utilizatori să folosească alt program. În cazul surselor deschise, dacă programul este cu adevărat valoros, se va găsi cu siguranță cineva care să-l dezvolte în continuare.

- În practică, principiul surselor deschise este foarte avantajos când trebuie să operezi modificări, de multe ori banale, într-un program.
30 De pildă, am vrut, cum este normal, ca bibliografia acestei cărți să nu fie presărată cu cuvinte englezești inutile, precum *and* (și). Am folosit accesul la sursele programului care determină stilul bibliografiei pentru a obține rezultatul dorit.⁴⁹

⁴⁷Nu cred că este etic să-i forțăm să folosească programe comerciale scumpe. S-ar putea ca astfel să-i împingem pe calea pirateriei. De aici și până la producerea de eseuri plagiate nu mai este decât un pas. Dacă un gen de hoție este permis, alte tipuri de hoție devin și ele acceptabile.

⁴⁸De aceea un bun editor de texte este atât de important pentru programatori.

⁴⁹Un program care nu permite accesul la surse oferă posibilitatea de a regla o serie de parametri. Dacă parametrul care ne interesează nu este reglabil trebuie să ne adresăm direct firmei producătoare. Licența nu ne permite să modificăm programul.

A.2 Editorul Vim

Vim continuă tradiția principalului editor din lumea Unix, editorul vi. Vi a fost creat de William Joy.⁵⁰ În ciuda faptului că, în forma sa actuală, Vim dispune de meniuri, mulți s-ar putea să-l găsească neatrăgător.

Cred că unii se gândesc deja să-și scrie eseu cu unul dintre *office-uri*.⁵¹ Cartea de față vă propune însă altă perspectivă decât cea pe care o adoptă un program pentru munca de birou. Problema nu este de a avea un program pentru munca de birou care să fie gratuit.⁵² Ideea este de a schimba perspectiva utilizării cu aceea a programării.

**perspec-
tiva
progra-
mării** Din perspectiva programării, un document tipărit este realizat cu ajutorul unui program. Programul este cel care stabilește așezarea în pagină a textului, imaginilor, tabelelor. Tot prin program sunt specificate tipurile de litere folosite. Tot programarea este cea care ne permite să inserăm note, trimiteri, bibliografii.

A.2.1 Învățarea Vim într-o zi

Vim este mai prietenos decât vechiul vi. Vi te pune în fața unui ecran negru. Apăsai tastele și nu se întâmpla nimic, dacă nu tastai din greșeală i sau a. Atunci apăreau niște litere pe ecran, dar mare lucru nu se întâmpla. Te întrebai cine o fi folosind Vi. Răspunsul este relativ simplu: cei care scriu programe.

Mergeți la adresa de Internet <<http://www.vim.org>> pentru a

⁵⁰William Joy este unul dintre fondatorii Sun Microsystems. Vi ține însă de perioada premergătoare celei de la Sun. Pentru istoria Vi v. interviul luat lui William Joy de către Eugene Eric Kim, *Linux Magazine*, <<http://www.linux-mag.com>>, noiembrie 1999 (data accesării: 11/01/2004), „What inspired you to write vi?”.

⁵¹Are, evident, sens să instalați un *office*. Puteți pune foarte ușor, de pildă, OpenOffice, care este versiunea cu sursă deschisă a unui produs al firmei Sun, firma lui William Joy. Programul acesta poate face și operații care depășesc sfera operațiilor normale într-un birou. Este un program care, dacă mergeți pe ruta **Tools**→**Data Sources** veți vedea, cu titlu de demonstrație, cum puteți lucra și cu o bibliografie ale cărei elemente le puteți integra în documentul dumneavoastră. Apăsați **OK**. Apoi apăsați **F4**. Expandați arborele Bibliography și ramura Tables. Veți vedea chiar datele pe care le cuprinde bibliografia demonstrativă instalată o dată cu programul. OpenOffice are și o serie de alte facilități pentru munca intelectuală. De asemenea, este capabil să deschidă și să creeze documente într-o serie de formate întâlnite frecvent.

⁵²Dacă folosiți Vim, vedeți desigur că acesta nu este un program pur și simplu gratuit. Bram Moolenaar îi încurajează pe utilizatori să facă donații unei fundații care desfășoară activități caritabile în Uganda.

afla cum puteți descărca Vim de pe Internet. Versiunea cu care sunt scrise și sursele acestei cărți este 6.2.

Instalarea versiunii 6.2 sub Windows este cât se poate de simplă. Vă recomand să optați pentru o apropiere cât mai mare de stilul Windows. Astfel, ceea ce ați învățat exersând cu Notepad puteți aplica în continuare.

Optați, de asemenea, pentru posibilitatea de a recurge la Vim direct din meniurile contextuale atașate oricărui fișier.

A.2.1.1 Un editor modal

Vim este un editor dedicat în special editării de fișiere de tip text. Este bine să vă familiarizați mai întâi cu modul în care Vim vă permiteți să prelucrați un fișier de tip text.

Creați un fișier de tip text în 2xExplorer. Deschideți-l cu Vim, nu cu Notepad. Dacă ați reușit, vedeți cum clipește un dreptunghi negru în colțul din stânga sus al ferestrei Vim. Inutil apăsați diverse litere. De regulă, nu veți face decât să apară litera respectivă pe ultima linie a ecranului. Tastați **i**. Forma cursorului s-a schimbat. Pe ultimul rând al ferestrei scrie acum -- INSERT --, iar dumneavoastră ați început să descoperiți, fără să știți, ideile de bază ale programării.

La început erați în modul numit „modul normal“. Puteți reveni oricând în acest mod apăsând tasta **Esc** care este plasată în colțul din stânga sus al tastaturii unui PC. În acest mod Vim interpretează apăsarea oricărei taste ca pe o comandă sau ca pe începutul unei comenzi.

A.2.1.1.1 Atenție la moduri Când au fost introduse calculatoarele la bordul avioanelor, un teribil accident a avut loc în Franța din pricina nefamiliarizării piloților cu ideea de **mod de lucru**. Piloții au vrut să ocolească un munte. Au dat o comandă pentru a câștiga în altitudine. Din păcate, nu au fost atenți la **modul** în care se afla programul calculatorului de bord. Avionul s-a dus cu încăpățănare direct spre munte.

Când deschideți Vim vă aflați în modul normal. Dacă tastăm **d** de două ori, ștergem un rând! Din fericire, putem tasta **u** și rândul se reface.

Sistemul complet al modurilor Vim este descris în documentația care însoțește editorul.

- A.2.1.1.1.1 Modul normal** Modul normal al Vim este cel în care dați comenzi dintr-un punct de pe suprafața de lucru a editorului. Cea mai frecventă comandă în acest mod va fi probabil **i**. Tastând **i** puteți începe să introduceți text.
- i**
- a** Încercați să vedeți care este efectul apăsării diverselor taste în modul normal. Care este, de pildă, diferența tastarea lui **a** și a lui **i**? Care este efectul lui **A**? 5
- A.2.1.1.1.2 Inserarea textului** Felul în care ajungem să introducem text a fost deja explicat (vezi secțiunea A.2.1.1, rândul 19). Ca să ieșim din acest mod apăsăm tasta **ESC** și revenim în modul normal. 10
- A.2.1.1.1.3 Înlocuirea unei porțiuni de text** Din modul normal puteți trece în modul în care suprascrieți textul deja existent în două feluri: dacă tastați **r** sau dacă apăsați combinația de taste **SHIFT+R**. În primul caz, după înlocuirea unei litere reveniți la modul normal. În al doilea caz, reveniți în modul normal doar apăsând **ESC**. 15
- r**
- Sub Windows, este probabil mai comod să suprascriem chiar pe parcursul introducerii de text. Apăsați tasta **INS**. Pe ultima linie a ferestrei Vim apare -- **REPLACE** --. Ca să ieșim din acest mod (familiar oricui a mai utilizat programe Windows!) apăsăm din nou tasta **INS**. 20
- A.2.1.1.1.4 Selectarea textului ca-n Windows** Pentru a selecta text țineți tasta **SHIFT** apăsată și folosiți tastele cu săgeți sau cele pentru derularea paginilor. Observați mesajul - (**insert**) **SELECT** - care apare pe ultimul rând al ferestrei Vim. 25
- În acest mod puteți lucra cu obișnuitele comenzi Windows de copiere, decupare și inserare de text.⁵³
- A.2.1.1.1.5 Modul vizual** Vim are și modul său propriu de a selecta text. Dacă, în modul normal, tastați **v**, atunci treceți în acest mod. Puteți selecta apoi text în maniera Windows sau folosind comenzi de deplasare a cursorului specifice Vim. 30
- V** Modul vizual are o versiune în care selectați câte un rând în întregime sa.⁵⁴ Pentru aceasta tastați **V** în modul normal.

⁵³Vezi secțiunea A.1.1.1.1.

⁵⁴Este vorba despre rânduri în sens logic. Vezi despre rânduri și secțiunea A.2.1.2.

La prima vedere, modul vizual nu prezintă mare interes. El are însă unele avantaje subtile în raport cu selecțiile Windows când trebuie să copiem text dintr-un mod Vim într-altul.

Regula pe care v-aș recomanda să o urmați este următoarea:
 5 folosiți cu selecțiile în stil Windows comenzile (de copiere, decupare, lipire) în stil Windows. Când vreți să folosiți comenzi specifice Vim, utilizați modul vizual pentru selecții.

Există, de asemenea, un mare avantaj al selecției în modul vizual: selecția unor coloane de text. Creați un tabel precum cel de la pagina 153, rândul 3. Mergeți în colțul uneia dintre coloane, treceți în modul normal și apăsați CTRL-Q. Acum puteți selecta coloana
 10 respectivă. **CTRL-Q**

În Unix selecția blocurilor de text în mod vizual se face după apăsarea CTRL-V, dar acest lucru nu este posibil sub Windows,
 15 unde combinația respectivă de taste servește la inserarea unui text copiat sau decupat.

A.2.1.1.1.6 Comenzile în linie Comenzile pe care le putem da tastând doar o literă sau folosind combinații de taste sunt fatalmente limitate. Vim are însă un set de comenzi (native) mult
 20 mai bogat. Pentru a avea acces la ele trebuie să treceți din modul normal în modul comandă în linie. Cum? Apăsând în modul normal pe tasta : (tasta cu două puncte pe ea).⁵⁵ **trecerea la comanda în linie**

În aceste anexe găsiți fragmente de cod. Fiecare rând este numerotat. Puteți obține și-n Vim o numerotare a rândurilor textului.
 25 În modul normal tastați :set nu! (nu vă speriați din pricina faptului că după ce-ați tastat două puncte treceți pe ultima linie; este semnul c-ați schimbat modul). Apăsați ENTER și apar numerele.

Cum scăpați de numere? Treceți iar în modul comandă în linie! Apăsați tasta cu săgeata în sus. Vim ține minte comenzile în linie.⁵⁶
 30 Repetați comanda anterioară. Numerele dispar.

⁵⁵Se poate trece în modul comandă în linie și apăsând tasta /. Acest lucru îl facem atunci când vrem să dăm o comandă de căutare prin restul textului (din punctul în care se află cursorul în modul normal). Tastăm însă ? (în modul normal) atunci când vrem să căutăm în porțiunile de text aflate înainte de punctul în care se află cursorul.

⁵⁶Vim ține minte separat comenzile puse după două puncte de cele realizate cu comenzile de căutare prin text.

A.2.1.2 Rânduri logice și rânduri vizuale

Scrieți, de pildă, ceva care vă trece pe moment prin cap. Apăsăți și Enter de câteva ori. Încercați să vă mișcați prin text cu săgețile de pe tastatură. Observați cum Vim consideră că un rând se termină logic când apăsați ENTER. Puteți să vă mișcați însă oriunde în text folosind mouse-ul și cursorul din Windows. Atenție însă! Pentru a schimba astfel poziția cursorului Vim trebuie să faceți clic în punctul în care vreți să vă opriți. 5

Distincția dintre rândurile logice și cele vizuale este foarte importantă. Rândurile vizuale depind de dimensiunea ferestrei Vim. Pentru a obține rânduri vizuale dați comanda `:set wrap!`; efectul ei este afișarea rândurilor logice în întregime lor, chiar dacă depășesc dimensiunile ferestrei. 10

Dezavantajul lui `wrap` este că rupe afișarea în mijlocul cuvintelor. Alternativa este comanda `:set lbr!`. Ea are drept efect crearea unor rânduri vizuale în care cuvintele rămân întregi. 15

A.2.1.3 Explorarea meniurilor

Citiți ceea ce găsiți când faceți clic pe **Help** în meniuri, dar nu încercați să pricepeți tot de la început. Pe de o parte, explicațiile par scrise pentru cineva destul de versat în lucrul cu acest gen de editoare. Pe de altă parte, multe dintre noțiuni sunt mai degrabă relevante în contextul unui sistem de operare de tip Unix. 20

Mai productiv, la început, mi se pare alt mod, foarte simplu de a învăța Vim. Parcurgeți cu răbdare meniurile programului și vedeți ce este de folos pentru dumneavoastră acolo. De exemplu, meniul File vă spune cum puteți deschide, închide și salva fișiere. 25

Veți vedea însă în meniuri și expresii misterioase de genul `:w`. Dacă scrieți `:w` și apăsați Enter veți salva fișierul și veți primi o confirmare de genul `"test.txt" 3L, 144C written`. Mesajul afișat de Vim cuprinde numele fișierului, numărul de rânduri, numărul de caractere și cuvântul englezesc *written*, pentru a ne indica scrierea pe disc a fișierului respectiv. 30

La meniul Edit acordați importanță în special secțiunii File Settings și rubricii Select Font. . . Mergeți la File Format. . . și observați că Vim cunoaște cele trei tipuri de fișiere text.⁵⁷ 35

Letterele (în engleză *fonts*) sunt o problemă cu mult mai dificilă.

⁵⁷Fișierele Dos sunt totuna cu fișierele text din Windows. Vezi mai sus (pagina 127, rândul 28) deosebirea dintre trei tipuri de fișiere text.

Alegeți un tip de literă (în engleză, *font*) care dispune de un Script de tipul Central European. În acest fel aveți acces la literele românești. Nu trebuie decât să comutați în Windows pe tastatura românească.⁵⁸ Vim este perfect capabil să lucreze cu ea. litere românești

5 Dacă revenim acum la meniurile principale, urmează trei meniuri mai ciudate: Tools, Syntax, Buffers. Nu este nevoie să le studiați cu foarte mare atenție de la bun început. Primul dintre cele trei meniuri va fascina desigur pe ucenicii vrăjitori. Cu ajutorul lui puteți învăța, de pildă, cum să editați codurile numerice pe care le folosește Windows în fișierul dumneavoastră. Al doilea ne introduce o trăsătură crucială a Vim: capacitatea sa de a vizualiza în culori diferite comenzile și datele din fișiere.⁵⁹ Al treilea trebuie studiat cu atenție. Conceptul folosit acolo nu trebuie confundat cu cel de ferastră. Buferele sunt însă efectiv utile când prelucrăm deodată un mare număr de fișiere.

A.2.1.3.1 Configurarea Vim Cum putem regla modul în care se comportă Vim? Pentru a înțelege acest lucru trebuie să ne amintim că Vim vine din lumea Unix. În acest sistem de operare obiceiul este să se folosească fișiere de configurare. Dacă mergeți în dosarul în care este instalat Vim veți găsi un fișier care se numește `_vimrc`.⁶⁰ Imediat după instalare veți găsi rândul următor în acest fișier:

```
1 source $VIMRUNTIME/vimrc_example.vim
```

Căutați scriptul `vim` la care se face referire în acest rând în dosarul în care se află principalul fișier executabil din Vim `gvim.exe` și faceți o copie a acestui fișier. Redenumiți copia. Cel mai normal ar fi să-i ziceți simplu `vimrc.vim`; puteți acum să faceți modificările dorite în această copie. Apoi, în `_vimrc` modificați în mod adecvat rândul care trimitea la exemplul de configurare creat la instalare.

Pe calculatorul cu care este scrisă cartea adăugirile mai importante sunt următoarele:

```
1 set guifont=lucida_console:h12:cEASTEUROPE
2 set listchars=tab:\|_,trail:-,eol:<
3 set lbr!
```

⁵⁸V. §B.3.1

⁵⁹Pentru detalii v. §A.2.1.3.3.1.

⁶⁰Linia din fața numelui fișierului imită stilul Unix. În Unix fișierele de configurare au nume care încep cu un punct. Linia n-are un mare rol aici. S-ar putea să fie o măsură de precauție față de programele antivirus care văd un dușman în tot ce nu este ca-n Windows.

```
4 set shiftwidth=3
5 set tabstop=3
6 set noexpandtab
```

literele Prima linie de mai sus spune Vim ce font să folosească atunci când deschide un fișier, cât de mare să fie corpul literei și să recurgă la un litere printre care se găsesc și cele românești. Rolul celorlalte linii îl puteți descoperi lesne dacă folosiți bara cu instrumente (bara

ajutor cu pictograme) și faceți clic pe semnul de întrebare cu lupă. Puneți

Vim cuvântul care vă interesează în caseta de dialog și cereți ajutor. 5

Atenție! Semnul de exclamare, egalul etc. nu fac parte din cuvintele-cheie ca atare.

A.2.1.3.2 Scripturile Vim Vim are desigur și multe puteri ascunse. Este imposibil să convertești în meniuri tot ce poate face Vim. 10

limbajul Importantă este extensibilitatea sa. Vim folosește pentru aceasta

Vim propriul său limbaj.⁶¹ Acest limbaj nu este însă foarte greu de înșușit și putem crea cu ajutorul lui o mulțime de lucruri utile, inclusiv noi meniuri.

script În limbajul Vim se pot scrie programe. Un program Vim este scris într-un fișier de tip text, dar care are extensia `.vim`. Aceste fișiere sunt colecții de comenzi pe care le interpretează Vim. Există și un nume generic pentru astfel de programe. Ele se numesc „script”-uri. 15

vimfiles În primul rând, merită să extindeți Vim cu ajutorul altora. În funcție de modul în care ați instalat Vim, căutați un dosar care se numește `vimfiles`. Căutați în el dosarul `plugin`. Recomandarea ar fi să nu folosiți dosarul `plugin` din `vimfiles`. Puteți astfel distinge între extinderile operate direct din instalare și cele ulterioare. O altă variantă ar fi să păstrați unul dintre dosare pentru extinderile create de alții și una pentru cele create de dumneavoastră. 20 25

Eu am adăugat, de pildă, `UnMtchBracket.vim`. Este o extensie creată de Chandra Naveen. Îți permite să sesizezi dacă o paranteză a fost sau nu închisă. După cum veți observa ulterior, acest lucru este absolut esențial din perspectiva programării. Extensia aceasta vă scutește de multe bătăi de cap. 30

Uneori vreți să vedeți însă care este perechea unei paranteze date. Pentru aceasta procedați în felul următor. Treceți în modul comandă. Puneți cursorul pe paranteza căreia vreți să-i găsiți perechea. Apăsăți `%`. Cursorul se va muta pe paranteza pereche. Puteți face acest lucru cu toate cele trei tipuri de paranteze. 35

⁶¹Spre deosebire de Emacs, care folosește Lisp.

O altă extindere utilă a Vim este `DirDo.vim`. Autorul acestui program este William Lee. Valoarea sa o veți descoperi imediat ce încercați să modificați foarte multe fișiere aflate în dosare diferite.

A.2.1.3.3 Vim bun la toate Ucenicii vrăjitori vor vrea desigur să deschidă cu Vim printr-un dublu clic o mulțime de fișiere. Pot face acest lucru în felul următor. Se pornește programul Regedit. În `HKEY_CLASSES_ROOT` este localizată cheia `txtfile`. Se merge pe ramura care pornește din această cheie până se ajunge la comanda pentru deschiderea fișierelor de tip text și acolo se schimbă comanda care spune Windows să folosească Notepad cu una care deschide fișierul cu Vim (ceva de genul `D:\use\vim\vim62\gvim.exe %1`). Evident, calea către `gvim.exe` trebuie să fie cea de pe calculatorul respectiv. Trebuie pus și `%1`, care spune sistemului să deschidă fișierul pe care se face dublu clic.

Nu trebuie însă abuzat de procedurile de genul celei de mai sus. Putem deregla relativ lesne Windows. De asemenea, de multe ori este mai interesant să deschidem simultan mai multe fișiere cu Vim sau să deschidem cu Vim un anume tip de fișier numai atunci când vrem să edităm fișierul, nu și atunci când vrem să-l vizualizăm.

A.2.1.3.3.1 Verificarea corectitudinii sintactice Fiecare limbaj de programare are sintaxa sa. Greșelile de sintaxă trebuie evident eliminate. Pentru a ne ajuta în procesul de identificare a erorilor de sintaxă Vim colorează textul programelor.

Comanda pentru colorarea textului este dată în fișierul de configurare. Vim folosește extensia fișierelor pentru a detecta limbajul în care este scris conținutul lor. De asemenea, Vim folosește informațiile conținute în primul rând al anumitor fișiere.

Dacă Vim nu detectează sintaxa sau vreți să o porniți manual, mergeți la meniul **Syntax**; apăsați **Show filetypes** în menu pentru a putea alege limbajul dorit. Restul comenzilor au nume care explică limpede rolul lor.

**evidențierea
sintaxei
prin
colorare**

A.2.1.3.3.2 Integrarea Vim în 2xExplorer Din 2xExplorer poate se poate apela direct la un editor extern. Folosind **View→Options...** bifați butonul radio pentru editorul extern și indicați calea către `gvim.exe`.

Puteți deschide acum cu Vim orice fișier prin simpla apăsare a tastei **F4**. Metoda aceasta este foarte comodă și este o bună alternativă la meniul contextual.

F4

A.2.1.3.3.3 Câte cuvinte sunt în fișier? Studentele și studenții sunt uneori foarte nedumeriți când li se cere ca eseul lor să nu depășească un anumit număr de cuvinte. Vor să măsoare totul în pagini de text.

În epoca fișierelor electronice este însă relativ lesne să numeri 5
cuvintele.

Vreți să aflați câte cuvinte ați scris în fișier? Treceți în modul comandă. Tastați G. Apoi tastați CTRL+G (țineți tasta CTRL apăsată și, în același timp, apăsați G). Pe ultimul rând al ferestrei Vim va apărea un mesaj care vă spune la al câtelea cuvânt sunteți și câte 10
cuvinte sunt în fișier.⁶²

A.2.1.3.3.4 Semne de carte În fișierele foarte mari este greu să revenim cu ușurință la un punct din fișier. Vim are posibilitatea de a se deplasa la rândul dorit, dar pentru a folosi această metodă trebuie să notăm numere de rând. Mai simplu este să punem 15
semne de carte.

Treceți în modul comandă (modul normal al Vim). Tastați m. Apoi tastați o literă oarecare. Este recomandabil ca litera respectivă să vă fie cumva asociată cu tema textului în punctul unde am pus 20
semnul de carte.

Mergeți în altă parte a textului. Cum reveniți în locul în care am pus semnul de carte? Treceți în modul comandă. Tastați '. Tastați 25
apoi litera care slujește drept semn de carte și veți reveni în punctul în care ați pus semnul de carte.

A.2.2 Vim la modul serios 25

Ce facem, de fapt, cu Vim? Am tot vorbit despre fișierele text. Eu am crea unul pe care l-am folosit când am testat exemplele care urmează. Este un fișier foarte simplu, cu următorul conținut:

```
1 filosofie uman  
2 \section{woo} poveste mit  
3 filosofic bibliografie
```

Am botezat acest fișier woo.txt. Litera w sugerează faptul că este vorba de un fișier text de tip Windows. Puteți crea lesne unul la fel 30
sau unul asemănător.

⁶²Pentru mai multe detalii a se vedea manualul scris de Bram Moolenaar pentru utilizatorii Vim (secțiunea 12.5).

Mergeți în meniul principal la **T**ools și dați clic pe **C**onvert to HEX. Fereastra Vim se modifică. În cazul fișierului meu `woo.txt` conținutul ferestrei arată astfel:

```
1 0000000: 6669 6c6f 736f 6669 6520 756d 616e 0d0a  filosofie uman..
2 0000010: 5c63 6861 7074 6572 7b77 6f6f 7d20 706f  \section{woo} po
3 0000020: 7665 7374 6520 6d69 740d 0a66 696c 6f73  veste mit..filos
4 0000030: 6f66 6963 2062 6962 6c69 6f67 7261 6669  ofic bibliografi
5 0000040: 650d 0a                                     e..
```

5 Ce s-a întâmplat? Vim a folosit un program extern⁶³ pentru a ne dezvălui din ce anume este făcut fișierul `woo.txt`. Fișierul este pur și simplu o colecție de coduri.

xxd

10 Rezultatul afișat de către Vim constă din cinci rânduri. În stânga vedeți numerele pe care le-am pus aici pentru a vă ușura numărătoarea. În centrul primelor patru rânduri se văd opt grupuri de patru cifre. Trebuie să grupați aceste cifre câte două. Fiecare pereche de cifre reprezintă un număr scris în baza 16.⁶⁴

S-ar putea să vi se pară curios, dar scrierea numerelor în baza 16 este mai comodă. De ce? Procesorul computerului lucrează, de fapt, cu numere în baza 2. El vede miezul unui tabel de genul următor:

```
1 00: 01100110 01101001 01101100 01101111 01110011 01101111  filoso
2 06: 01100110 01101001 01100101 00100000 01110101 01101101  'fie um
3 0c: 01100001 01101110 00001101 00001010 01011100 01100011  an..\c
4 12: 01101000 01100001 01110000 01110100 01100101 01110010  hapter
5 18: 01111011 01110111 01101111 01101111 01111101 00100000  {woo}
6 1e: 01110000 01101111 01110110 01100101 01110011 01110100  povest
7 24: 01100101 00100000 01101101 01101001 01110100 00001101  e mit.
8 2a: 00001010 01100110 01101001 01101100 01101111 01110011  .filos
9 30: 01101111 01100110 01101001 01100011 00100000 01100010  ofic b
10 36: 01101001 01100010 01101100 01101001 01101111 01100111  ibliog
11 3c: 01110010 01100001 01100110 01101001 01100101 00001101  rafie.
12 42: 00001010
```

15 Observați că nu mai sunt acum decât șase coloane în centrul tabelului. Sunt aceleași numere ca și mai sus, dar scrise în baza 2. Se văd limpede grupurile de opt cifre de 0 sau 1. În limbajul informaticii un grup de asemenea cifre binare se numește „octet”⁶⁵

Creați cu ajutorul Vim un fișier `woof.txt` cu următorul conținut:

⁶³Programul `xxd.exe`, pe care-l puteți folosi independent de Vim. Îl găsiți în dosarul în care se află și `gvim.exe`.

⁶⁴De aici denumirea de „hex” (de la *hexadecimal*).

⁶⁵În engleză, *octet* sau *byte*.

```
1 filosofie uman
2 \section{woof} poveste mit
3 filosofic bibliografie
```

În 2xExplorer selectați pe rând cele două fișiere și apăsați tasta F12. Comparați dimensiunile! Fișierul `woo.txt` are 67 de octeți(*bytes*); fișierul `woof.txt` are 68 de octeți. Fișierele diferă între ele printr-o singură literă! Litera `f` de pe rândul 2 diferențiază cele două fișiere.

octet Concluzia este foarte simplă. Un octet servește la codificarea unei litere. Dar un octet este format din două grupuri de patru 0 sau 1. În reprezentarea în baza 16 a octetului fiecărui grup de patru cifre binare îi corespunde o singură cifră. Este mai comod să folosim scrierea codului numeric în baza 16: scriem doar două cifre, nu opt. Iar trecerea este directă: fiecare cifră din *hex* este tradusă separat în patru cifre binare corespunzătoare. 5 10

Uitați-vă și la numărătoarea de pe margine. Noi am folosit obișnuita reprezentare în baza 10 a numerelor pentru rânduri. Programul `xxd` a pus însă (pe coloana separată cu două puncte de rest) cifre în baza 16. În baza 16, 10 din baza 10 este `a`, 11 este `b`, 12 este `c`, 13 este `d`, 14 este `e` și 15 este `f`. Când avem de a face cu o reprezentare în baza 16 formată din două cifre, prima cifră trebuie înmulțită cu 16 pentru a-i afla valoarea (în baza 10, mai accesibilă nouă); rezultatul, adunat cu a doua cifră ne dă, reprezentarea în baza 10. Uitați-vă pe rândul opt mai sus: cifra `2a` reprezintă numărul $2 * 16 + 10 = 42$ (în baza 10). Este numărul de octeți de pe primele șapte rânduri. 15 20

Complicat? Și da și nu. Ar fi imposibil să manevrăm direct codurile numerice. Vim ne oferă însă o vizualizare a acestor coduri. Programul ne ajută, de asemenea, să facem tot felul de modificări în coduri: să adăugăm, să ștergem, să schimbăm între ele coduri și așa mai departe. 25

Să ne concentrăm atenția asupra codurilor în baza 16. Puteți să identificați codul pentru litera `f`? Este `66`.⁶⁶

spațiu alb are un cod Un spațiu alb are și el un cod! Este vorba de `20`. Dacă fișierele dumneavoastră nu coincid cu dimensiunile indicate aici, probabil aveți vreun spațiu alb pe undeva. 30

⁶⁶Citiți „șase-șase“! Nu este chiar corect, dar „șazeci și șase“ ar fi absolut greșit, deoarece este vorba despre o reprezentare în baza 16 a numărului pe care în baza 10 îl reprezentăm prin 102. „Șase ori șaisprezece plus șase“ ar fi corect, dar – dacă n-aveți cumva capacitatea de a face rapid calcule în minte – mai mult ne încurcă.

Sfârșitul de rând este indicat prin două coduri (în Windows). Acestea sunt Od, zis CR, și 0a, zis LF.⁶⁷

Există un cod pentru sfârșitul de fișier? În principiu da, dar Vim a construit fișierele fără să introducă acest cod. Nu este o greșeală.

- 5 Codul respectiv ar fi 1a, zis și EOF⁶⁸. Vim l-ar vizualiza (în modul în care este setat la instalare) cu un ^Z.

Există firește un cod și pentru bara oblică inversă. Acest cod este 5c.

- 10 În rezumat, serviciul fundamental pe care ni-l face un editor ca Vim este acela de a ne facilita manipularea colecțiilor de coduri de genul celor descrise mai sus. Vim nu este destinat afișării frumoase a unui text într-o pagină. Nu este un sistem de tipărire, fie pe ecran, fie pe hârtie.

A.2.2.1 Atracția interfețelor grafice

- 15 Vi este editorul favorit al persoanelor care folosesc **metoda oarbă** de tastare.⁶⁹ Pentru a folosi **oarba** trebuie însă să ții permanent degetele pe taste.⁷⁰ Mouse-ul mai mult încurcă lucrurile pentru că trebuie să ridici mâna de pe taste.

- 20 Vim continuă tradiția vi și poate fi folosit fără a ridica mâna de pe taste. În versiunea sa cu interfață grafică Vim are însă meniuri și permite recursul masiv la serviciile mouse-ului.

- În continuare, vom folosi meniuri pentru a prezenta posibilitățile Vim. În general, vom urma ordinea temelor din manualul lui Bram Moolenaar[29]. Moolenaar explică însă ideile pentru cineva
25 care folosește cu prioritate tastatura.⁷¹

A.2.2.1.1 Construirea meniurilor Vim are un sistem foarte flexibil de adăugare a unor meniuri suplimentare.⁷² Pentru a învăța scrierea de meniuri, creai un dosar special pentru exerciții. În acest dosar (folder) creai un fișier aidoma lui `woo.txt`⁷³.

⁶⁷Pentru detalii despre sfârșitul de rând în diverse tipuri de fișiere vezi §A.1.2, pagina 127, rândul 28.

⁶⁸De la expresia englezească *end of file*.

⁶⁹Numele sugerează faptul că persoana respectivă se uită la ecran sau în altă parte, dar nu la taste.

⁷⁰Tastele F și J au pe ele mici profile distincte pentru a putea reveni lesne la poziția standard a mâinilor pe tastatură.

⁷¹Pentru explicații cuprinzătoare, bogat ilustrate, vezi cartea despre Vim scrisă de Steve Oualline[32].

⁷²A se vedea manualul lui Moolenaar[29, §42].

⁷³Vezi pagina 152, rândul 28

Comenzile pentru toate meniurile pe care le veți crea trebuie
un prim puse în fișiere cu extensia **vim**. Creați, în dosarul pentru exerciții,
menu un fișier **simplicu.vim**; aici veți pune cel mai simplu meniu posibil.
Scrieți următoarea linie de cod (fără numărul de linie din față!):

```
! :imenu Simplicu.text Am realizat un prim meniu.
```

Salvați conținutul fișierului.

Deschideți fișierul **woo.txt** cu ajutorul Vim. Sunteți în modul normal! Mergeți cu ajutorul mouse-ului sau al tastelor cu săgeți pe ultimul rând. Apăsați tasta **O** (fără să apăsați **SHIFT**!).⁷⁴ Cursorul se mută pe rândul următor și editorul trece în modul **insert**. Pe bara cu instrumente (bara cu pictograme) dați clic pe omulețul care aleargă (**Run a Vim Script**). Se deschide o casetă de dialog. Deschideți **simplicu.vim** ca pe orice fișier. În bara cu meniuri apare un nou meniu.

Acum puteți da un clic pe meniul **Simplu** și apoi pe **text**. Pe ultimul rând al fișierului apare textul **Am realizat un prim meniu**. Puteți da un clic pe pictograma **Undo** (săgeata răsucită către stânga) și anula efectul meniului dumneavoastră. Cu un alt clic pe **Undo** aduceți fișierul la forma inițială.

Ce trebuie să studiați în codul meniului? În primul rând cele trei blocuri ale meniului.

Primul bloc începe cu două puncte. Am putea deci folosi acest cod pentru o comandă în linie! Apoi urmează **imenu**, care spune programului că este vorba de un meniu care funcționează în modul **insert**.

Al doilea bloc conține textul care apare în meniuri. Atenție la punct. Nu puneți spații albe aici! Ar fi interpretate ca un semn că urmează al treilea bloc. Dacă vreți spații albe, trebuie să le precedați cu o bară oblică inversă, ca-n exemplul următor:

```
! :imenu Simplicu.pre\ text Am realizat un prim meniu.
```

În al treilea bloc pot să apară spații albe. Al treilea bloc conține comenzile pe care le execută Vim când faceți clic pe **item**-ul respectiv din meniu. În exemplele de mai sus suntem în modul **insert** și ceea ce i se spune, de fapt, editorului Vim este să insereze textul respectiv în fișier.

⁷⁴Combinatia **SHIFT+O** creează un rând deasupra celui pe care am dat comanda.

A.2.2.1.1.1 Submeniuri Dacă mergeți la meniul standard Edit și apoi la File Settings, vedeți cum apare un submeniu. Puteți exersa crearea de submeniuri în Vim meșterind liniile de cod care urmează:

```
1 :imenu Simplu.nod.text1 Am realizat submeniul 1.
2 :imenu Simplu.nod.text2 Am realizat submeniul 2.
```

- 5 Fiți atente și atenți la faptul că trebuie să dați din nou clic pe pictograma cu omulețul care aleargă! Vim trebuie ținut la curent cu schimbările pe care le-ați făcut.

Dacă vreți să luați totul de la început, închideți fișierul text. Redeschideți apoi fișierul și repetați operațiile de creare a meniului.

- 10 **A.2.2.1.1.2 Acceleratori** Mai mult ca sigur, știți deja la ce slujesc literele subliniate din titlurile meniurilor. Combinația de taste ALT+E, de pildă, deschide meniul Edit. Apoi este suficient să apăsați tastele care corespund literelor subliniate pentru a avansa prin meniu sau pentru a executa comanda asociată respectivului
- 15 articol din meniu.

Cum pot fi realizați acceleratorii? Foarte simplu. În linia de cod de mai jos `m` este transformat într-un accelerator.

```
1 :imenu Si&mplu.&accelerat Am realizat un meniu accelerat.
```

Puneți semnul `&` în fața literei care desemnează acceleratorul.

- 20 Trebuie să fiți atente și atenți la posibilele conflicte cu alte meniuri sau inconsistențe. Dacă nu folosiți des acceleratorii, atunci n-are probabil rost să-i creați.

A.2.2.1.1.3 Bara cu instrumente B. Moolenaar explică în manual cum poate fi modificată bara cu instrumente.⁷⁵

- 25 Efectul obținut prin modificarea barei cu instrumente vă va convinge probabil că Vim este efectiv flexibil. Dacă aveți pasiunea interfețelor grafice, veți pune desigur cel puțin o pictogramă proprie pe bara cu instrumente.

Să vedem un exemplu practic. Creați un fișier `vim` care conține următorul cod:

```
1 :tmenu ToolBar.LtrCmd Comandă LaTeX
2 :amenu ToolBar.LtrCmd <Esc>\{}<Left><Left>
```

⁷⁵Moolenaar[29, §42.4].

Prima linie de cod spune editorului ce mesaj să afișeze în mica fereastră care apare când ducem mouse-ul pe pictograma de pe bara cu instrumente. A doua linie include comanda. Am pus **amenu** pentru ca pictograma să fie afișată în toate modurile Vim⁷⁶. Comanda ca atare este în cel de-al treilea bloc. Ea îi spune lui Vim să înseze scheletul unei comenzi \LaTeX , format din semnele $\{\}$ și apoi să mute cursorul după bara oblică inversă. Pe **<Esc>** îl puteți ignora deocamdată. Nu-l eliminați însă.⁷⁷

La urmă, dar nu în cele din urmă, examinați blocul din mijloc al celor două linii de cod de mai sus. Scrieți **ToolBar** și nu altceva. Nu schimbați nici majusculele în minuscule sau invers. Altfel Vim nu înțelege că vreți să puneți ceva pe bara sa cu instrumente. Vă trebuie însă și o imagine pentru pictograma de pe bară.

Eu am creat un dosar (un folder) **bitmaps** în dosarul standard **vimfiles**. Vim citește o variabilă de mediu care-i spune unde sunt dosarele sale.⁷⁸ Dosarul **bitmaps** trebuie să fie undeva unde caută Vim. Eu prefer **vimfiles** pentru că aici stau doar fișierele create sau instalate local (de către mine) și nu cele din instalația standard Vim. De asemenea, nu schimbați numele lui **bitmaps**. Puneți în **bitmaps** imagini de tip **bmp** cu dimensiunea de 18×18 pixeli.

Imaginile din dosarul **bitmaps** sunt folosite pentru pictograme. Numele lor trebuie să fie exact același cu cel care urmează după **ToolBar** și punct. În exemplul nostru, imaginea trebuie să aibă desigur numele **LtxCmd.bmp**.

Dacă nu știți nimic despre imaginile folosite de către programele de computer, puteți afla câte ceva aici în §C.1.

A.2.2.1.1.4 Meniurile standard

Morala de bază în cazul meniurilor standard este să nu le modificați. Dați meniurilor dumneavoastră alte nume. Folosiți alte taste pentru accelerare.

Morala aceasta fundamentală nu vă interzice însă să examinați felul în care sunt construite meniurile standard. Mergeți în dosarul **Vimxx**.⁷⁹ Căutați fișierul **menu.vim** și studiați-l.

Nu folosiți unele dintre părțile meniurilor standard? Nu folosiți,

⁷⁶ Acel a din **amenu** vine de la englezescul *all modes* (toate modurile).

⁷⁷ Citiți despre caracterul aparte al lui **amenu** în manualul lui Moolenaar[29, p.207]. Practic, acel **<Esc>** este echivalent cu apăsarea tastei **ESC**, necesară pentru a anula efectul lui **CTRL-O** introdus automat în modul **insert** de către **amenu**. Noi vrem doar să introducem un text, nu să executăm o comandă și abia apoi să revenim la modul **insert**.

⁷⁸ La instalare sunteți întrebat unde vreți să puneți **vimfiles**.

⁷⁹ ele două x -uri stau pentru numerele versiunii Vim.

de exemplu, din `Tools` secțiunea a treia, cea care se referă la compilare? Nu este nevoie să distrugeți ceva în `menu.vim`. Eu am pus în fișierul de configurare al Vim următoarele rânduri:

```
1 :aunmenu Tools.&Make
2 :aunmenu Tools.&List\ Errors
3 :aunmenu Tools.&List\ Messages
4 :aunmenu Tools.&Next\ Error
5 :aunmenu Tools.&Previous\ Error
6 :aunmenu Tools.&Older\ List
7 :aunmenu Tools.&Newer\ List
8 :aunmenu Tools.Error\ Window
9 :aunmenu Tools.Set\ Compiler
```

Aceste comenzi fac să dispară din meniu porțiunile pe care nu le folosesc. n-am distrus însă nimic pentru a putea restaura lesne meniurile inițiale.

Atenție la comenzile de mai sus. Ele au între litera care indică modul Vim și `menu` secvența `un`. Restul se înțelege de la sine.

Mai există o metodă blândă de schimbare a meniurilor standard.

10 Putem transforma toată linia de cod (de program deci) într-un *commentariu*. Vim ignoră comentariile. Comentariile sunt rânduri (în sens logic, nu vizual) care încep cu ghilimele duble. Exemplul următor ne arată cum scap de linia de separare rămasă în plus în meniu după comenzile de anulare de meniuri de mai sus:

**comenta-
rii în
Vim**

```
1 "an 40.520 &Tools.-SEP3-<Nop>
```

15 Singura modificare pe care am făcut-o a fost inserarea ghilimelelor duble la început de rând. Evident, este ușor să le elimin la nevoie.

Tot cu ghilimele duble am scăpat și de unele pictograme:

```
1 " an 1.250 ToolBar.Make :make<CR>
2 " an 1.270 ToolBar.RunCtags :!ctags -R .<CR>
```

A.2.2.2 Mișcările cursorului Vim

Cu un simplu clic pe butonul din stânga mouse-ului putem schimba poziția cursorului în fereastra Vim. Foarte comod! Cătuși de puțin dacă vrem să scriem un meniu Vim care să facă și altceva decât să insereze text.

Vim este un editor, după cum am văzut, de colecții de coduri pentru tot felul de semne. În aceste condiții ne interesează prea puțin punctele de pe ecran și afișarea pe ecran a semnelor. Ne-ar interesa să ne deplasăm cumva de la un cod la altul. Este ceea ce explicăm în continuare.

A.2.2.2.1 Mișcări ale cursorului în modul normal În modul normal, care este modul în care putem da comenzi apăsând pe taste, există o serie de taste care controlează mișcările cursorului. Este destul să indicăm aceste taste într-un script Vim pentru ca să obținem același efect ca atunci când apăsăm fizic pe taste.

5

A.2.2.2.1.1 Micile mișcări Dacă ați studiat felul în care se creează meniurile, nu este greu să folosiți următoarele rânduri într-un script Vim:

```
1 menu Cursor.<- h
2 menu Cursor.^ k
3 menu Cursor.v j
4 menu Cursor.-> l
```

După cum se observă mai sus, cele două puncte din fața cuvântului `menu` nu sunt absolut necesare. Evident dacă ați folosi tastatura, ar trebui să apăsați pe tasta cu două puncte pentru a trece în modul comandă în linie.

10

Meniul pe care-l creați nu este, în practică, de mare folos. După cum sugerează și semnele noastre, săgețile pot face lesne același lucru. Importantă este însă descoperirea efectelor comenzilor.

15

A.2.2.2.1.2 Numărarea pașilor Ne putem însă mișca nu doar cu pași de melc. Putem face un fel de salturi de cangur.

```
1 menu Cursor.<- 3h
2 menu Cursor.-> 3l
```

Cifra din fața literei indică de câte ori se aplică comanda. Fișierul meu `woo.txt` este prea strâmt pentru salturi peste rânduri, dar ideea se aplică și-n acel caz.

20

Meniurile nu se compară aici ca eficiență cu tastarea directă în modul normal, dar jucăria ar trebui să poată sugera posibilitățile de acțiune ale Vim. Eu unul mă joc pornind de la litera `w` de pe rândul 2 din `woo.txt`.

A.2.2.2.1.3 De la un cuvânt la altul Deplasarea de la un cuvânt la altul este o chestiune mult mai subtilă. Pentru început puteți studia posibilitățile următoarelor meniuri:

25

```
1 "comanda w deplasează cursorul la următorul început de cuvânt
2 menu Cursor.w w
```



```

3 "comanda e deplasează cursorul la următorul sfârșit de cuvânt
4 menu Cursor.e e
5 "comanda b deplasează cursorul la precedentul început de cuvânt
6 menu Cursor.b b
7 "comanda ge deplasează cursorul la precedentul sfârșit de cuvânt
8 menu Cursor.ge ge

```

Comentariile de pe rândurile 1, 3, 5 și 7 oferă explicațiile necesare. Restul este o chestiune de exercițiu.

Ca și-n cazul pașilor mici, se pot și aici număra pașii care urmează să fie făcuți. Putem da înapoi, de pildă, cu trei cuvinte.

- 5 **A.2.2.2.1.4 Deplasările pe distanțe lungi** Putem să mergem însă de la un capăt la altul al rândului. Putem merge de la un capăt la altul al fișierului.

```

1 "comanda ^ deplasează cursorul la început de rând (logic)
2 menu Cursor.capRand ^
3 "comanda b deplasează cursorul la sfârșit de rând
4 menu Cursor.sfarsitRand $
5 "comanda gg deplasează cursorul la început de fișier
6 menu Cursor.capFisier gg
7 "comanda G deplasează cursorul la sfârșit de fișier
8 menu Cursor.sfarsitFisier G

```

- 10 Asemenea deplasări sunt foarte utile într-un script Vim. Putem spune deja că scripturile Vim sunt programe veritabile. Uneori vrem, de pildă, să căutăm ceva prin tot fișierul. Atunci are sens să mergem la începutul fișierului și să începem căutarea.

- 15 **A.2.2.2.2 Mișcările cursorului în modul insert** În scripturile Vim putem comanda mișcările cursorului și fără să fim în modul normal (cel în care dăm comenzi). Metoda este limpede dacă veți construi un meniu folosind rândurile de cod Vim de mai jos:

```

1 imenu Cursor.Stanga <Left>
2 imenu Cursor.CuvantStanga <C-Left>
3 imenu Cursor.Sus <Up>
4 imenu Cursor.Jos <Down>
5 imenu Cursor.Dreapta <Right>
6 imenu Cursor.CuvantDreapta <C-Right>
7 imenu Cursor.CapRand <Home>
8 imenu Cursor.SfarsitDeRand <End>

```

Meniul este activ în modul insert.

Se observă lesne că denumirile sunt mai sugestive (pentru cei care știu limba engleză). Nu omiteți parantezele unghiulare! Ele fac parte din limbajul în care sunt scrise programele Vim.

O explicație aparte necesită doar <C-Left>. Ideea este că C ne arată că ținem tasta CTRL apăsată. Similar, S-Left conține o indicație de apăsare a tastei SHIFT. Evident, în programe, aceste apăsări de taste sunt apăsări virtuale, nu reale.

Comenzile <Left>, <Up> și celelalte pot fi folosite și-n modul normal. Ele pot fi precedate de un contor al pașilor doar în modul normal. În modul insert comanda 3<Left> ar avea drept rezultat inserarea cifrei 3 în text.

A.2.2.2.1 Trecerea de la un mod la altul Pentru a trece de la un mod la altul putem folosi în comenzi pe <Esc>. De pildă, dacă suntem în modul normal, <Esc>i sau <Esc>a fac trecerea la modul insert. Invers, pentru a trece din modul insert în modul normal, putem folosi pe <Esc>.

A.2.2.3 Modificarea fișierelor

Putem modifica un fișier prin adăugarea de text. Putem modifica un fișier prin inserarea de text. Asta este tot ce știm dacă stăpânim doar mișcările prin fișier.

Pentru a copia porțiuni de text sau pentru a le șterge, pentru a înlocui porțiuni de text ar trebui să stăpânim alte tehnici.

A.2.2.3.1 Registrii de memorie Presupunerea noastră constantă este că Vim este configurat pentru lucrul în maniera MS Windows. Oricine a inspectat bara cu instrumente din Vim, a remarcat trei pictograme care servesc la modificarea textului: o foarfecă, două foi de hârtie și un clipboard. Se pot folosi, de asemenea, combinațiile CTRL+X CTRL+C CTRL+V pentru a obține aceleași efecte ca și prin clicuri pe pictogramele amintite.

Clipboard-ul standard din Windows are însă o mare limită. Nu ține minte decât o singură porțiune de text o dată. Are, ca să zicem așa, o memorie de scurtă durată.

Creați însă un meniu Vim folosind următoarele rânduri de cod Vim:

```
1 "Copie în registrul a
2 menu Redactor.copieA "ay
```

```

3 "Scrie din registrul a
4 menu Redactor.scrieA "ap
5 "Copie în registrul b
6 menu Redactor.copieB "by
7 "Scrie din registrul b
8 menu Redactor.scrieB "bp
9 "Copie în registrul c
10 menu Redactor.copieC "cy
11 "Scrie din registrul c
12 menu Redactor.scrieC "cp

```

Eu m-am jucat cu fișierul `woo.txt` în felul următor: am creat un rând suplimentar gol; am selectat cuvântul „bibliografie”; meniul `Redactor` s-a activat și am copiat textul selectat în registrul (sectorul) de memorie `a`; am pus apoi cuvântul „filosofic” în registrul de memorie `b` și cuvântul „woo” în registrul `c`. Am folosit linia liberă pentru a plasa acolo conținutul celor trei regiștri de memorie. Fiți atenți însă la faptul că meniul `Redactor` este activ doar în modul normal.

Să analizăm acum comenzile din meniu. Apoi trebuie să ne lămurim ce rol au ghilimelele duble. Ele trebuie folosite în modul normal (modul comenzilor) pentru a-i spune lui Vim să aștepte restul comenzii. De pildă, dacă am selectat text în modul vizual al Vim, putem da o comandă de genul `"dy`. Vim pune porțiunea selectată în registrul `d`. Cum? Acel `y`, care vine de la termenul *yank* este cheia comenzii respective.⁸⁰

yank

Ce regiștri putem folosi? Vim are nouă tipuri de regiștri.⁸¹ În practică, la început veți folosi probabil regiștrii numerotați, de la `0` la `9`, și regiștrii care au drept nume o literă, de la `a` la `z`, respectiv de la `A` la `Z`. În orice caz, cel mai simplu este să puneți text în regiștrii care au drept nume o literă.

A.2.2.3.2 Copiere inteligentă Programarea se învață cel mai bine meșterind. S-ar putea ca ucenicul vrăjitor să se întrebe dacă nu-r putea renunța la ghilimelele duble. Nu în meniurile de mai sus. Cu selecțiile am putea lucra, dar, în modul normal, un `a`, de pildă, ne-ar duce în modul insert. Vim nu-r ști că este vorba despre registrul de memorie. Și totuși ucenicul vrăjitor are dreptate.

Există o soluție. Meșteriți meniuri cu ajutorul următoarelor rânduri de cod Vim:

```
1 "Copie un cuvânt
```

⁸⁰Pentru copiere vezi manualul lui Moolenaar[29, §4.6].

⁸¹Vezi în documentația Vim fișierul `change.txt` pentru o descriere completă.

```
2 menu Redactor.copieCuvant bye
3 "Copie (inclusiv) până la acolada din dreapta
4 menu Redactor.copiePanaLaAcoladaDreapta yf}
```

Dacă *yank* n-are o selecție cu care să lucreze, atunci va aștepta o comandă care să-i spună ce să copieze.

Mergeți în *woo.txt* la prima acoladă și copiați până la acolada din dreapta cu ajutorul meniului. Apăsăți tasta P în modul normal pentru a vedea rezultatul. Observați că s-au copiat și acoladele. Cum? Secvența *f}* este o comandă care-i cere editorului Vim să găsească⁸² acolada din dreapta.

Dacă nu vreți și acoladele, puneți cursorul pe cuvântul „woo” și copiați cu ajutorul meniului cuvântul. Puteți apoi să vă gândiți cum este realizată comanda.

Hm, mormăie nemulțumit ucenicul vrăjitor. Dacă nu este un singur cuvânt între acolade! Atunci folosește:

```
1 "Copie până la acolada din dreapta
2 menu Redactor.copiePanaLaAcoladaDreapta vf}hy
```

Selecția textului este făcută chiar în interiorul comenzii.

A.2.2.3.3 Inserare de text în stil Vim Dacă *yank* este un termen pe care nu-l auzim în lumea Windows, în schimb p-ul folosit pentru a pune text amintește de *paste*-ul din Windows.

atenție la diferența Vim distinge însă între două feluri de a pune text, după cum se vede mai jos:

dintre

minuscule

```
1 "Plasează (după cursor) ceea ce ai copiat
```

```
și 2 menu Redactor.plaseazaDupa p
```

majuscule

```
3 "Plasează (înainte de cursor) ceea ce ai copiat
```

```
4 menu Redactor.plaseazaInainte P
```

Toată diferența este dată de folosirea unei litere minuscule sau a uneia majuscule.

A.2.2.3.4 Decupare/eliminare de text în stil Vim Vim poate șterge o singură literă. Bănuiesc însă că v-ar interesa mai mult eliminările masive de text. Meșteriți meniurile următoare și veți vedea ce poate să facă Vim:

⁸²În limba engleză *find*.

```

1 "Ștergem o literă
2 menu Sterge.x x
3 "Șterge până la sfârșitul cuvântului
4 menu Sterge.dw dw
5 "Șterge până la sfârșitul celui de al treilea cuvânt
6 menu Sterge.d3w d3w
7 "Șterge un rând
8 menu Sterge.dd dd
9 "Anulează efectul comenzii anterioare
10 menu Sterge.undo u
11 "Anulează anularea unei comenzi
12 menu Sterge.redo <C-R>

```

Am adăugat și capacități de a elimina dezastrele produse de ștergeri sau anulări de ștergeri.

Cum decupăm? Foarte simplu. Mergeți la începutul unui cuvânt și ștergeți (decupați) cu ajutorul comenzii `dw` din meniu. Mergeți în alt punct din fișier și, în modul normal, apăsați tasta `P`.

Ce facem dacă vrem să decupăm un cuvânt fără să mergem la începutul său? O soluție este:

```

1 "aw este text-object (adică un cuvânt)
2 menu tObiect.unCuvantSters daw

```

Subiectul este însă cam avansat pentru ambițiile acestei anexe.⁸³

A.2.2.4 Meniuri care fac viața mai ușoară

10 Meniurile de până acum au avut mai mult un rol pedagogic. Ca atare, nu sunt foarte utile. La ce este însă cu adevărat bun un meniu? Este bun atunci când ai de tastat o secvență mai complicată de comenzi sau una simplă, dar pe care ai uitat-o.

15 Mai țineți minte cum se numără cuvintele din fișier? Probabil că nu. S-ar putea să găsiți meșterirea următoarelor meniuri utilă sau chiar foarte utilă:

```

1 "Numără cuvintele din fișier
2 menu Util.numara g<C-G>
3 "Aflați în ce loc din fișier vă aflați
4 menu Util.loc :set noruler<CR><C-G>
5 "Ca să puneți ruler-ul la loc
6 menu Util.ruler :set ruler<CR>
7 "Găsirea perechii unei paranteze
8 menu Util.perecheaParantezei %
9 "Schimbă minusculele în majuscule și invers

```

⁸³ A se vedea pentru detalii manualul lui Moolenaar[29, §4.8].

```
10 menu Util.schimbaCasaDeLitere ~
11 "Schimbă un cuvânt întreg;
12 "mai bine cu aw decât cu bve~
13 menu Util.schimbaCasaDeLitereCuvant vaw~
14 "Schimbă între ele două litere
15 menu Util.xp xp
```

Meniurile de pe rândurile 4 și 6 vă permit să vă aflați locul în care sunteți în fișier. Meniul de pe rândul 6 repară eventuala eroare produsă de meniul de pe rândul 4, dacă vă place să aveți permanent *ruler*-ul în fața ochilor.

perechea parantezei Găsirea perechii unei paranteze este ceva vital când folosim programe precum L^AT_EX. S-ar putea să socotiți utilă instalarea permanentă a scriptului `matchit.vim`. Îl găsiți în dosarul `macros` din distribuția standard a Vim.⁸⁴ 5

casa de literă Schimbarea casei de literă⁸⁵ este adesea o problemă. Această anexă am prezentat Vim dintr-o perspectivă elementară. Cunoștințele de bază de aici vă permit să editați însă fișiere de tip text simple. Problemele specifice fișierelor `tex` sau `bib` vor fi tratate separat. De asemenea, căutarea într-un fișier va fi discutată în alte anexe. 10

A.3 Expresiile regulate

 15

Dincolo de orice teorie sofisticată, ideea care stă în spatele expresiilor regulate poate fi lesne explicată printr-un exemplu. Să zicem c-am ieșit în oraș și studiez numere de înmatriculare ale mașinilor. Unele numere de înmatriculare încep cu una-două cifre și continuă cu o literă. Dar sunt vechile numere! Altele au o literă urmată de mai multe cifre. Par a fi numere provizorii. Majoritatea mașinilor au numere de înmatriculare noi. Cum le disting? Aș putea oferi o descriere de acest gen: la început sunt una sau două litere, apoi două cifre, urmate de trei litere. 20

Aș putea oare caracteriza cumva formal un număr de înmatriculare? Care este gramatica adecvată? 25

Mă pot gândi la numerele de înmatriculare ca la niște expresii. Fiecare expresie conține numere și litere (majuscule). Există o mulțime a tuturor acestor expresii. Toată problema constă în separarea

⁸⁴Instalarea ca atare este banală. Este bine să puneți `matchit.vim` în dosarul `plugin` din `vimfiles`. Instalarea ajutorului pentru acest script este descrisă în manualul lui Moolenaar[29, §5.5].

⁸⁵Acesta este termenul tehnic din poligrafie. Vine de la franțuzescul „casse”.

din mulțimea mare a acestor expresii a acelor expresii care constituie mulțimea numerelor de înmatriculare (posibile).

În ciuda posibilelor asociații greșite pe care le poate sugera numele, o „expresie regulată“ nu este o **expresie** în sensul de mai sus, ci un **tipar** (un șablon) care surprinde **regularitățile** care disting o expresie de alte expresii.

Din explicația de mai sus decurge ideea că unei expresii regulate îi corespunde o mulțime de expresii. Expresia regulată este un șablon care ne permite să decidem dacă o expresie aparține sau nu mulțimii respective. Aceasta este ideea care stă la baza gramaticilor cu expresii regulate.

apartenența la o mulțime de expresii

La ce ar fi bune aceste șabloane? În primul rând, fără ele nici nu poate fi vorba de căutări mai sofisticate prin fișiere. De asemenea, orice program de prelucrare de texte, de pildă, nu poate trece de un stadiu foarte rudimentar dacă n-are posibilitatea să distingă între diverse expresii.

A.3.1 Definirea șabloanelor

Unii scriu „filosofie“, alții „filozofie“. Eu unul nu cred că este o problemă de substanță în spatele acestei diferențe. Exemplul este însă bun pentru a explica modul de căutare într-un text cu ajutorul unor șabloane sau tipare.

Creați un fișier care cuprinde următorul text:

```
filosofie filozofie bibliofil
fiilosofie fiiilozofie b37fil
fiiiilosofie fiiiilozofie
filoşofie flosophie f+losophie
```

Un nume sugestiv pentru acest fișier ar „test.txt“.

Pentru o primă căutare în fișierul de mai sus folosim un tiparul următor:

```
/filo[sz]ofie/
```

Cu tiparul se potrivesc atât filosofie, cât și filozofie. Parantezele drepte indică faptul că oricare dintre literele din paranteză s-ar putea potrivi. Barele oblice sunt necesare în Vim pentru căutări și-nlocuiri de șiruri de semne. Le vom folosi în continuare și pentru că marchează clar granițele unui tipar.

bara oblică

Deschideți fișierul **test.txt** în Vim. Puneți cursorul la începutul fișierului.⁸⁶ Treceți în modul comandă în linie și tastați tiparul de mai sus. Când execută comanda, Vim mută cursorul la prima apariție a unui șir de semne care corespund tiparului și evidențiază cu ajutorul fondului colorat tot ce corespunde cu tiparul dat.⁸⁷

Vim folosește tipare pentru a filtra un text, separând ceea ce corespunde tiparului de ceea ce nu corespunde. Mulțimea expresiilor care corespund tiparului este evidențiată prin colorare.

A.3.1.1 Logica șabloanelor

Persoanele pasionate de logică vor descoperi cu plăcere modul de construire a tiparelor. În exemplul de mai sus este deja prezentă **variabilă** ideea de **lacună** într-un text. Astfel concep logicienii ideea de **variabilă**. Formal, o variabilă este o lacună dintr-un text; lacuna respectivă poate fi umplută cu un anumit material. Mai sus umplerea era limitată la literele **s** sau **z**. Puteam scrie **[a-z]** și atunci domeniul de valori al variabilei ar fi cuprins literele de la **a** la **z**.

punctul Dacă punem însă un simplu punct, tiparul devine **/filo.ofie/** și Vim va evidenția și cuvântul „filoșofie”. Punctul pus într-un tipar corespunde cu ideea de lacună pentru un semn. Tiparul **/f.l/** nu va produce nici o evidențiere în „flosografie”. Nici tiparul **/f[.]l/** nu va evidenția nimic! Ar trebui să avem în text pe „f.losografie” sau ceva de acest gen pentru ca să se producă o evidențiere.

Din nou, persoanele interesate de logică pot să mediteze la **cuantificarea** implicată de parantezele drepte puse în tipar. În cuvinte aceasta ar însemna *pentru oricare dintre semnele din paranteză*.

\? Punctul pus în tipar cere ca lacuna să fie completată efectiv. Pentru a admite și zero apariții ale unui semn trebuie, în setarea care este dată prin instalarea Vim,⁸⁸ trebuie scris un tipar de genul **/f\?losografie/**.

\+ Bara oblică spre stânga din **\?** nu ține de construcția standard a tiparelor. Ea depinde de modul în care este setat Vim. Bara îi spune lui Vim că nu trebuie să ia în sens literal semnul care urmează. Observați diferența dintre **/f+losografie/** și **/fi\+losografie/**.

Dacă vreți un mod standard de formare a tiparelor, atunci tre-

⁸⁶ Pentru că modul căutării va fi înainte. Pentru căutări înapoi ar trebui să folosim un semn de întrebare în locul primei bare oblice.

⁸⁷ Atenție! Verificați în meniul **Edit la Global Settings** dacă este dată comanda de activare-dezactivare a evidențierii.

⁸⁸ Ucenicii vrăjitori care au modificat **setările standard** ale Vim trebuie să consulte documentația pentru a putea obține efectele dorite cu tiparele folosite.

buie să puneți o opțiune \v chiar la începutul tiparului. Experimentați cu /\vfi+losofie/. Pentru alte opțiuni care afectează tiparele sau pentru eventuale nepotriviri cu explicațiile date aici citiți documentația Vim. **modul standard**

- 5 Am introdus deja mai sus o altă cuantificare în tipare. Ea ar putea fi citită în felul următor: una sau mai multe apariții ale semnului anterior. Astfel, tiparul /\vfi+losofie/ caută „filosofie“ cu una sau mai multe apariții ale lui i.

Cuantificarea * funcționează ca și +, dar admite și situația când în lacuna aflată în raza sa de acțiune nu a fost completată cu nici un semn. Setarea normală a lui Vim⁸⁹ ne permite să ne dispensăm de opțiune \v în acest caz. *

Putem cuantifica și numeric. Tiparul /\vfi{1,3}losofie/ va conduce la filtrarea acelor cazuri în care i apare cel puțin o dată sau cel mult de trei ori. Putem omite unul dintre numere. **cuantificare numerică**

Ce facem însă dacă, folosind un tipar standard, vrem să-l căutăm chiar pe +. În acest caz, vom scrie, de exemplu, /\vf\+losofie/. Evident, opțiunea \v am pus-o de dragul Vim. Oricum, în Vim, cum am văzut mai sus, putem să-l recunoaștem pe + direct.

- 20 Dacă vrem să găsim barele oblice într-un text, atunci trebuie să folosim \[, respectiv \/.

Tiparele de căutare pot fi foarte sofisticate. Putem indica, de exemplu, dacă tiparul trebuie căutat la început de rând. Pentru aceasta scriem /\v~fil/. Observați diferența față de /\vfil/. **început de rând**

- 25 Din nou, folosit între [] semnul ~ are cu totul alt rol. Dacă scriem /\[~0-9]/, aceasta înseamnă că suntem în căutarea a orice altceva în afară de cifre. Dacă exeperimentați cu fișierul test.txt, vedeți cum Vim subliniază tot, în afară de „37“. Spațiile sunt și ele semne! **excludere**

- 30 Dacă vrem doar cuvintele care încep cu f, trebuie să construim un tipar de genul /\v<f[a-z]>/. Practic cuvântul este pus în paranteze unghiulare. Dacă nu punem \v, atunci, în Vim, trebuie să punem bara oblică spre stânga ca în \< și \>. Când experimentăm observăm că „filofofie“ nu este evidențiat. Iar + trece drept graniță a unui cuvânt.⁹⁰ **cuvinte**

- 35 **A.3.1.1.1 Conectori pentru șabloane** Ca și-n logica propozițiilor, putem folosi conectori pentru obține șabloane compuse. Un

⁸⁹Ucenicii vrăjitori se pot juca folosind comenzile :set magic și :set nomagic; setarea recomandată este magic, dar nici ea nu este, cum am văzut mai sus, suficientă pentru o utilizare standard a tiparelor.

⁹⁰Cu titlu de exercițiu, puteți încerca să aflați ce efect are /\v<f[a-z,ç]>/.

tipar poate avea, de pildă, două ramuri, precum `/\vfil|biblio/`. Bara verticală o putem citi „sau”. Dacă experimentăm acest tipar pe fișierul `test.txt`, observăm desigur cum cuvântul „bibliofil” este evidențiat în întregimea sa.

Trebuie să fim atenți la spații. Spațiile puse în jurul barei verticale vor fi interpretate ca fiind o parte a tiparului. Dacă ni se pare confuză expresia, putem pune paranteze rotunde în jurul tiparelor conectate.

A.3.2 Utilizarea șabloanelor

Dacă nu scriem programe avansate, în practică, n-avem nevoie de structuri ale tiparelor mai complicate decât cele descrise mai sus.

substitu-
ție La ce am putea, de altfel, folosi tiparele, în afară de căutări în fișiere? Tiparele sunt utilizate frecvent în operațiile de substituie a unui semn sau șir de semne cu alt semn sau șir de semne. De pildă, pot înlocui pe „37” cu „73” în fișierul `test.txt` cu ajutorul următoarei comenzi: `:%s/37/73/`. Se observă acel `s` pus în fața tiparului șirului care va fi substituit, precum și felul în care este plasat șirul cu care se substituie.⁹¹ Semnul `%` spune lui Vim să caute în tot fișierul.

Comanda `:%s/\vfi+1/fil/` nu duce chiar la efectul scontat! Vim caută în tot fișierul, dar nu substituie peste tot. Pentru aceasta trebuie să punem după șirul cu care se substituie o opțiune. Dacă vrem o substituie globală, punem `g`. Comanda noastră este atunci ceva de genul `:%s/\vfi+1/fil/g`. Putem apăsa `ESC` și experimenta. Pentru a anula efectul comenzii apăsați, în modul comandă, `u` sau faceți clic pe pictograma `Undo`⁹².

Acum puteți face un prim pas către programare. Programarea nu este, în fond, decât arta de a proiecta algoritmi, de a descoperi pașii care duc la rezolvarea unei probleme. Fie următoarea problemă: în `test.txt`, substituiți pe „s” din „filosofie” cu „z” și pe „z” din „filozofie” cu „s”. N-ar fi rău să nu vă uitați o vreme la soluția care urmează și să experimentați propriile idei.

Exercițiul nu este chiar banal dacă n-ați scris niciodată programe. Dacă substituiți direct pe „z” cu „s” sau invers, ajungeți precum hoții derutați de Morgiana în *O mie și una de nopți*. Hoții au făcut un semn pe casa unde stătea Ali Baba, dar Morgiana l-a reprodus pe toate casele din cartier. Nu mai știți unde a fost „z” și unde a „s”.

⁹¹Evident, la substituie trebuie să indicăm un șir (posibil vid), nu tiparul unui șir. De altfel, dacă punem tiparul, Vim va substitui cu tiparul ca atare.

⁹²Numele pictogramelor apare scris sub ele dacă deplasați cursorul Windows pe pictograma respectivă.

Soluția este foarte simplă. Efectuați o substituie intermediară. **substi-**
De exemplu, dați comanda :%s/filosofie/filoÇofie/g. Apoi dați **tuire**
comanda :%s/filozofie/filosofie/g. Pasul al treilea, cel final, îl **interme-**
reprezintă :/filoÇofie/filozofie/g. **diară**

5 A.3.3 Utilizarea șabloanelor în programul grep și-n TeXnicCenter

Dacă vă întrebați ce programe, în afară de Vim, folosesc tipare pen-
tru căutări, cel mai simplu ar fi să instalați o unealtă Unix nu-
mită „grep“ sau să folosiți mediul integrat de dezvoltare de pro-
grame L^AT_EX.⁹³ Mergeți cu ajutorul 2xExplorer în dosarul în care
țineți **test.txt** și chemați fereastra MS-DOS. Tastați următoarea **grep**
comandă în fereastra MS-DOS: **grep filo[sz]ofie ***. Comanda are
trei părți: numele comenzii, un tipar după care se face căutarea și
numele fișierului în care se face căutarea. În locul numelui fișierului
am pus ***** pentru a arăta că este vorba despre orice fișier (din dosarul
respectiv).

Rezultatul este afișarea numelui fișierului în care grep a găsit
ceva care corespunde tiparului și a rândului în care apare șirul care
corespunde tiparului. Dacă vreți și numărul rândului, puneți op-
țiunea **-n** după **grep**. Pentru a vă convinge că grep caută efectiv
prin mai multe fișiere, creați un alt fișier text în dosarul respec-
tiv și scrieți acolo, între altele, „filosofie“ și/sau „filozofie“. Dacă
vreți instrucțiuni ceva mai detaliate pentru grep, dați comanda
grep --help > grep.txt. Această comandă va crea un fișier cu
explicații. Se observă c-am folosit pe **>**, care redirectionează rezulta-
tul comenzii. Acesta nu mai este afișat pe ecran, ci este scris într-un
fișier.

În TeXnicCenter căutați pictograma cu binoclul așezat peste un **TeXnic-**
dosar (Find in files). Când dați clic pe această pictogramă apare **Center**
o fereastră de dialog. Ar trebui să nu vă fie deloc greu să alegeți
tipul fișierelor. De asemenea, este relativ ușor să alegeți dosarul
unde se face căutarea (**D**irectory). Bifați butonul de validare **Regular**
expression.

Eu am avut, de exemplu, nevoie să caut contextele în care apărea
una dintre formele expresiei „ucenic vrăjitor“. Am folosit șablonul
ucenic[i]* vr. jitor[i]*. De ce? Voiam să găsesc și cazurile când
expresia era folosită la plural; de asemenea, era mai simplu să nu
mă încurc cu diacriticele.⁹⁴

⁹³Pentru instalarea TeXnicCenter vezi aici §B.1.2.

⁹⁴Puteți însă pune litere cu diacritice în șabloane. Atâta doar că trebuie să

TeXnicCenter construiește o listă cu toate contextele în care apare expresia. Dând clicuri pe lista respectivă deschideți fișierul chiar în punctul dorit. Acest mod de căutare de tip **grep** este unul dintre marile avantaje ale TeXnicCenter.

În TeXnicCenter puteți construi două liste cu contextele în care găsiți două expresii diferite. Puteți face comparații și lucra eficient cu fișierele aflate într-un dosar sau chiar în subdosarele acestuia.

A.4 În căutarea surselor

În mod tradițional răsfoiai cărți în bibliotecă. Este o plăcere la care eu unul nu aş putea renunța. Dar astăzi căutăm mai degrabă în biblioteci electronice și mai ales în imensa bibliotecă electronică denumită Internet.

Să zicem c-am găsit pe Internet *Frații Karamazov* sub forma unui fișier text.⁹⁵ Deschid fișierul în Vim. Dau un clic pe pictograma cu o foaie și o lupă pe ea. Apare o mică fereastră. Scriu cuvântul **Inquisitor** și apăs ENTER. Vim derulează textul și apare cuvântul-cheie pe care l-am tastat anterior pe un fond galben. Este chiar punctul din text în care Ivan îi spune lui Alioșa că a scris un poem despre Marele Inchizitor.

Evident, sunt foarte interesat de aspectele filosofice ale poemului lui Ivan. Pe moment, trebuie remarcată însă deosebirea principală din perspectiva căutării prin textul romanului: nu încerc să găsesc o pagină (acest concept nici n-are sens aici), ci locul în care apare un cuvânt-cheie.

căutare cu comandă în linie Pot căuta însă și cu ajutorul comenzilor în linie. Apăs tasta Esc urmată de / și cuvântul-cheie **Peter**. Cursorul se mută ceva mai jos în text în punctul în care este vorba despre Petru cel Mare.

chirilice În principiu, în textul în limba rusă se caută în același fel. Ceea ce complică lucrurile sunt literele chirilice. Când deschid textul apar semne incompreensibile. Merg la meniul **Edit** și apoi la **Select Font** pentru a schimba *script*-ul tipului de literă în Cyrillic. Acum pot citi textul. Mă mulțumesc să selectez un cuvânt din text și să-l copiez în fereastra de căutare a Vim, dar numai după ce am dus cursorul la începutul textului. Vim ascultă silitor comanda și colorează fondul cuvântului respectiv prin tot textul.

comutați pe tastatura cu literele cu diacritice înainte de a deschide caseta de dialog.

⁹⁵Romanul lui Dostoievski este disponibil în rusă; de asemenea, putem găsi traducerea în engleză realizată de Constance Garnett.

**selecția
vizuală
Vim**

Limitele lucrului cu ferestrele Windows se fac însă rapid simțite. Este mult mai sigur să procedez după cum urmează. Apăs ESC și apoi v. Vim scrie pe ultimul rând **VISUAL**, semn că sunt în modul lui specific de selecție. Țin tasta SHIFT apăsată și selectez cu săgeata către dreapta ca-n Windows. Când termin de selectat, copiez tot ca-n Windows selecția (prin clic pe pictograma cu cele două foi suprapuse). Trec în modul comandă în linie, titez / și lipesc textul copiat anterior, tot ca-n Windows, cu clic pe pictograma corespunzătoare. Căutarea funcționează. Pot să repet căutarea trecând în modul comandă în linie și recurgând la istoria comenzilor păstrată de Vim (apăs săgeata îndreptată în sus).

Pot scrie cuvântul-cheie căutat direct în linia de comandă, dar pentru aceasta trebuie să dispun în Windows sau Vim de tastatura pentru limba rusă.

Puteți afla mai multe despre căutările cu ajutorul Vim citind *help*-urile. Aici n-avem spațiu decât pentru discuția de principiu.

A.4.1 Construirea unei concordanțe

Este limpede că nu toți cei care se ocupă cu filosofia au și o pasiune specială pentru logica formală. Mai mult decât atât, s-ar putea să aibă nevoie de căutări amănunțite prin fișiere persoane care studiază texte vechi sau stilul unor filosofi. A le cere să folosească în practică tipare construite de la un capăt la altul de către ele ar fi probabil o utopie.

Pe de altă parte, din ceea ce știu din propria practică, atunci când studiezi un text filosofic ai nevoie de o listă cu aparițiile unui cuvânt-cheie în text. Dacă lista aceasta oferă și contextul în care apare cuvântul-cheie, ai deja ceea ce se numește o „concordanță”.

Puteți găsi pe Internet un program gratuit de realizare de concordanțe scris de Zdenek Martinek și Les Siegrist. Numele programului este „Wconcord”.⁹⁶

Instalarea programului Wconcord este cât se poate de simplă. Dacă l-ați găsit arhivat, îl dezarhivați într-un dosar potrivit și puteți deja lucra. n-ar mai fi necesară decât o scurtătură într-un dosar din Start Menu. Puteți crea, de exemplu, un dosar pentru Editare și-n el unul pentru Analiza textelor.

Wconcord construiește concordanțele pe baza fișierelor de tip

⁹⁶ A se vedea subsecțiunea A.4.2 pentru o explicație a modului în care puteți găsi programul pe Internet.

text.⁹⁷ Prima operație care trebuie făcută cu programul este elaborarea unei liste a fișierelor în care se vor opera căutările.

frecvența
cuvintelor

Programul are o interfață grafică și este extrem de ușor de folosit. Cel mai simplu lucru este să construim o listă a cuvintelor care apar în text și să o ordonăm în funcție de frecvență. Am experimentat acest lucru cu textul romanului lui Dostoievski *Frații Karamazov*. Aliașă, de exemplu, este menționat de 1243 de ori. Termenul „moarte“ apare de 121 de ori, în vreme ce „libertate“ apare de 53 de ori.

5

concor-
danțe
pentru
docu-
mente
pdf

Mai interesante decât frecvențele sunt utilizările în context ale unor termeni-cheie. Am folosit practic acest sistem pentru a analiza textul tratatului lui Ludwig von Mises *Human Action*. Textul disponibil pe Internet este în format pdf. Am extras cu ajutorul Adobe Acrobat Reader textul pur și am folosit sistemul de investigare oferit de Wconcord. Regăsirea punctului din text în formatul pdf nu este o problemă dacă folosim în mod adecvat un fragment-cheie de text. Astfel putem stabili cu exactitate toate referirile unui autor la un anumit concept.

10

15

Foarte interesante sunt concordanțele în care este identificat contextul în care apar mai mulți termeni-cheie. Wconcord poate lucra simultan cu cinci termeni-cheie. Putem specifica și alternative la un termen-cheie. De asemenea, poate fi precizată ordinea în care apar cheile și distanța maximă dintre ele.

20

Wconcord poate lucra, de asemenea, cu formele gramaticale ale unui cuvânt. Pentru aceasta este nevoie să construim noi fișierele cu lista de forme gramaticale.

25

A.4.2 Arheologie pe Internet

Toate programele despre care este vorba în această carte sunt disponibile pe Internet. Este deci firesc să spunem câteva cuvinte despre modul în care putem găsi un document pe Internet.

30

Presupun că nu scrieți eseul și efectuați căutările pe Internet pe același calculator. Puteți merge în laboratorul Facultății și veți găsi acolo o rețea de calculatoare care este conectată la Internet. De altfel, Internetul nu este decât o rețea de rețele de calculatoare care acoperă tot globul. Fiecare rețea mai mică are un calculator care le deservește pe celelalte. Termenul englezesc pentru acest calculator este *server*. Dacă rețeaua este mai dezvoltată s-ar putea să existe

35

⁹⁷Extragerea textului este posibilă din orice fișier. De la un format la altul, trebuie văzut însă cum se procedează și ce program poate să facă acest lucru.

un alt calculator care are instalate pe el programele care formează poarta către Internet. Termenul englezesc este *gateway*. Calculatoarele acestea nu se disting de celelalte neapărat prin modul în care construite, ci prin sistemul de operare de care dispun.

5 Categorie, Windows98 este nepotrivit pentru conectarea la Internet. Este vulnerabil la atacurile de pe Internet. Ceea ce-l face atât de plăcut când este folosit pe un calculator separat, ușurința cu care avem acces la fișiere devine un handicap când persoane rău intenționate au și ele cam același acces facil la fișiere.

10 Oricine poate face pe ucenicul vrăjitor și vedea cât de vulnerabil este Win98. Mergeți cu 2xExplorer pe discul C. Nu faceți însă modificarea care urmează decât dacă aveți organizarea fișierelor sugerată în acest capitol.⁹⁸ Redenumiți dosarul **Program Files** prin simpla eliminare a spațiului din nume. Veți primi un avertisment, dar atât.

15 Sistemul nu vă împiedică să acționați. Încercați acum să deschideți un fișier cu extensia **html**. Dacă n-ați modificat locul în care se află Internet Explorer, veți primi un mesaj care spune că sistemul nu poate descoperi unde se află **iexplore.exe**. Redenumiți **Program Files** pentru a-l aduce la forma pe care o știe sistemul.

e ușor să
distrugi;
greu e să
constru-
iești

20 Ați văzut deci că un pirat de pe Internet nu trebuie să fie prea inteligent. Este mult mai ușor să distrugi decât să construiești.

Calculatoarele cu Win98 sunt cel mult bune în rețea în postura de *clienți* ai *server*-ului. Acesta este cel care le servește cu ceea ce doresc și are misiunea de a le apăra de atacuri.

25 În laboratorul Facultății, sistemul de operare al serverelor este de tip Unix (GNU/Linux și Sun Solaris). Cea mai mare parte din clienți au și ei sisteme Unix. Pe acești clienți, pentru a vedea fișierele **html** tipice pentru Internet se folosește programul Netscape.

Un client Windows folosește în mod tipic programul Internet Explorer. În orice caz, sub Win98, puteți folosi acest program pentru a vizualiza fișiere de tip **html**. Acestea sunt tot fișiere de tip text, dar conțin o mulțime de comenzi folosite de programul de vizualizare pentru a produce imaginea frumoasă de pe ecran. Fișierele **html** conțin, de asemenea, trimiteri către alte fișiere aflate poate la mari

35 distanțe în spațiu.

În esență, chiar și atunci când vreți doar să vedeți ceva pentru o clipă trebuie să descărcați fișiere. Este ca și cum ați suna pe cineva la telefon, iar persoana respectivă v-ar dicta un text. Cam tot așa, un calculator apelează numărul unui alt calculator și își procură

40 fișierul de care este nevoie.

⁹⁸În orice caz, să n-aveți 2xExplorer în Program Files.

Ca persoană umană ar fi greu să ții minte numerele de pe Internet ale calculatoarelor. De aceea se folosesc adrese umanizate. De exemplu, adresa paginii de pe Internet a acestei cărți este `<www.fil.unibuc.ro/~solcan/eft/>`.

Dacă vorbești la telefon trebuie să știi și limba persoanei cu care conversezi. Tot așa, adresa de mai sus nu este complet funcțională fără o indicație privitoare la limbă. Completată astfel, ea devine `http://www.fil.unibuc.ro/~solcan/eft/`. 5

descărcarea de fișiere de pe Internet Să zicem că ai ajuns la pagina de Internet a cărții și ai descoperit un *script*, un program, pe care vrei să-l descarci pe calculatorul dumneavoastră. În Internet Explorer, poți duce cursorul Windows pe numele fișierului respectiv; obții apoi un meniu prin clic pe dreapta. Folosiți **Save Target As...** pentru a descărca fișierul. Dacă utilizați Netscape, procedura este asemănătoare, numai că recurgeți la **Save Link As...** 10 15

De multe ori veți găsi indicații cu privire la descărcarea de fișiere (numită în limba engleză *download*). Evident, citiți condițiile în care puteți obține fișierele. Citiți licențele aferente și avertismentele.

În tot acest proces cel mai important lucru este să ajungeți însă la pagina dorită. De unde aflăm ce să tastăm după acea magică *Address* din Internet Explorer? Există un fel de „grep“-uri ale Internetului. Ele se numesc „motoare de căutare“. Așa cum sugerează și numele, veți fi deservite și deserviți de o mașinărie. Cu puțină ingeniozitate din partea dumneavoastră, rezultatele vor fi însă din cele mai bune. 20

motoare de căutare Cartea de față oferă și ea adrese de pe Internet. Acestea se pot însă schimba. Este mult mai simplu și mai sigur să apălați la un motor de căutare. Tastați, de pildă, după *Address* sau *Location*, comanda `http://www.google.com` și citiți instrucțiunile de utilizare a motorului. 25

Tehnica de căutare se bazează pe cuvinte-cheie. Îi veți oferi motorului un șir de cuvinte-cheie și acesta vă va furniza adrese pe Internet și o serie de informații despre ceea ce găsiți la adresa respectivă. 30

Cygwin Pentru a găsi, de exemplu, **Cygwin** am folosit cuvintele-cheie `cygwin Linux emulation download`. Căutarea a durat 0,24 de secunde. Am primit, ce-i drept, o listă cu 3630 de adrese. Pare enorm, dar motorul le-a ordonat deja în mod automat. A doua adresă este `<http://www.cygwin.com/>`; a patra este `<http://www.redhat.com/download/cygwin.html>`. Pare rezonabil să citesc informațiile generale de la prima adresă și să află la a doua adresă cum pot descărca Cygwin. 35 40

Acum se vede de ce este mai rațional sistemul căutărilor decât

recursul la o singură adresă găsită în carte. Sunt destul de multe locurile de pe Internet unde putem găsi ceea ce căutăm. Atunci când locul respectiv este organizat sub forma unei colecții de fișiere și pentru transferarea lor se folosește limbajul **http**, în engleză, se vorbește despre un *site*. Cred că este potrivit să folosim în românește cuvântul „sit”. Folosim de multă vreme expresia „sit arheologic”. Aici este vorba de „sit pe Internet”. Căutarea pe un asemenea sit presupune câteodată veritabile înclinații pentru arheologie. Cele mai mici fragmente de informații trebuie exploatate. Vom ilustra acest lucru cu o serie de exemple.

Următorul exemplu este absolut crucial pentru cartea de față. De data aceasta căutăm un loc de unde să descărcăm o distribuție **T_EX**. Cuvintele-cheie folosite sunt doar **TeX distribution Windows98**. Informațiile despre a cincea adresă arată o legătură cu **MikT_EX**. Găsim astfel un posibil capăt de fir de care să tragem pentru a ajunge, navigând de la o pagină la alta, la <http://www.miktex.org/>, pagina de web a distribuției **T_EX** descrise și-n această carte.

Ucenicii vrăjitori nu vor rezista probabil tentației de a introduce cuvintele-cheie **C++ IDE SourceForge download**. Dacă vor căuta atent ce se potrivește cu Win98, vor descoperi adresa <http://www.bloodshed.net/devcpp.html>. Cei mai curioși se pot uita și la pagina <http://sourceforge.net/projects/dev-cpp/>.

Mai sunt câteva programe extrem de utile a căror căutare merită să o ilustrăm aici. Dacă folosim cuvintele-cheie **TeXnicCenter download**, găsim adresa de la care putem descărca mediul integrat pentru crearea de programe **L^AT_EX**: <http://www.toolscenter.org/products/texniccenter/download.htm>.

Dacă utilizăm cuvintele-cheie **weaverSlave HTML editor download**, descoperim adresa de Internet <http://www.subjective.de/en/weaverslave/index.php>. De aici se poate descărca un editor de fișiere **html**, **php** și comenzi **SQL**, foarte util pentru crearea de pagini web.

Dacă sunteți în căutarea unui program pentru statistică, găsiți o soluție cu ajutorul cuvintelor-cheie **ViSTA statistics**. Prima adresă din listă este chiar cea a paginii de Internet a Dr. Forrest Young, creatorul programului **ViSTA**, <http://forrest.psych.unc.edu/research/vista-frames/welcome.html>.

Un alt mod de a găsi documente sau programe pe Internet este reprezentat de paginile cu trimiteri către documente sau programe. Dacă navigați pe Internet la adresa <http://www.uni-giessen.de/~ga1007/ComputerLab/concordance.htm>, găsim o pagină în

care se explică pe scurt ce este o concordanță și o listă cu programe de creat concordanțe. Pagina aceasta nu oferă însă direct posibilitatea de a descărca programe. La <http://www.ujaen.es/dep/filing/profesores/alejandro_alcaraz.html> găsim o pagină care oferă această posibilitate. Puteți descărca direct arhiva Wconcord de la adresa <<http://www1.ujaen.es/~aalcaraz/HEL/wconcord.zip>>.

A.4.2.1 Programul wget

O adresă precum cea pentru Wconcord poate fi folosită și cu una dintre cele mai bune unelte Unix transpuse sub Windows, programul **wget**. Acest program poate fi apelat fie sub Cygwin, fie ca program independent, în funcție de modul în care l-ați instalat. Este un program cu comandă în linie. Are o serie de avantaje însă față de **Save Target As...** Nu trebuie să descarci interactiv zeci de fișiere care sunt menționate într-o pagină.

Comenzile pentru **wget** sunt de forma **wget [opțiuni] [adresa-pe-Internet] [opțiuni]**. Adresa pe Internet este opțională; putem folosi comanda în forma următoare: **wget -i nume-fișier**. În fișierul folosit vom scrie adresa sau adresele care ne interesează. Astfel nu trebuie să tastăm adresele în linia de comandă.

Opțiunea **-nc** împiedică descărcarea repetată a aceluiași fișier. O opțiune extrem de utilă, pe care o putem pune în finalul comenzii este **-k**; ea îi spune lui **wget** să convertească trimiterile. În loc ca trimiterile să fie făcute pe Internet, ele vor fi locale. Fișierul va putea fi consultat fără probleme și pe un calculator care nu este conectat la Internet.

Opțiunea **-p** este, de asemenea, utilă pentru a vedea pagina când nu suntem conectați. Ea îi spune lui **wget** să descarce tot ce este necesar (imagini, sunete) pentru a vedea pagina.

Opțiunile **-r** și **-l** trebuie folosite cu grijă pentru că ele vor pune **wget** să descarce fișiere în mod recursiv până la un anumit nivel. Aceasta înseamnă că **wget** descarcă o pagină, apoi se uită la ce trimite pagina respectivă și descarcă, până ajunge la nivelul la care i-ați spus să se oprească. Metoda are un dublu dezavantaj: poate suprasolicita calculatorul de unde vreți să obțineți fișiere, forțându-l pe administratorul acestuia să vă blocheze accesul; de asemenea, dacă nu sunteți prudenți, vă puteți trezi c-ați descărcat un volum uriaș de date care v-au lăsat fără spațiu pe discul propriului calculator (și fără bani, pentru că va trebui să plătiți furnizorul de servicii

Internet). Nivelul până la care merge **wget** trebuie reglat cu maximă grijă. De asemenea, un simplu **Ctrl+c** va stopa **wget**.⁹⁹

O precauție contra descărcării în exces de fișiere o reprezintă opțiunea **-A** urmată de o listă de extensii de fișiere (separate prin virgule). Acestea sunt singurele tipuri de fișiere care vor fi descărcate. Alternativ, puteți pune opțiunea **-R** și specifica tipurile de fișiere care nu vor fi descărcate.¹⁰⁰

Programul **wget** este extrem de util pe o rețea instabilă. În acest caz, descărcarea de fișiere se întrerupe frecvent. Rămânem cu o bucată de fișier. Fișierele de mari dimensiuni nu pot fi practic descărcate. Folosind opțiunea **-c** putem continua descărcarea din punctul unde s-a rupt fișierul. Continuarea descărcării în acest mod nu depinde numai de **wget**, ci și de sprijinul pentru această operație pe serverul de pe care descărcăm fișierul. Nu toate serverele oferă sprijin pentru continuarea unei descărcări întrerupte.¹⁰¹

**descărcare
în rețele
instabile**

A.4.2.2 Limitarea ariei căutărilor

Căutarea cu ajutorul cuvintelor-cheie este deosebit de eficientă. Un șir de cuvinte-cheie este asemenea unei conjuncții de condiții. Nu este nevoie să fii expertă sau expert în logică pentru a-ți da seama că în acest fel se limitează lista cu adrese pe care o obținem de la motorul de căutare și, implicit, aria căutărilor noastre prin această listă.

Există însă o problemă a cuvintelor-cheie. Ca orice cuvinte, acestea pot avea o doză de ambiguitate. Există, de pildă, un limbaj de programare numit **python**. Ce se întâmplă dacă folosim acest cuvânt pentru o căutare pe Google? Vor veni adrese cu situri despre limbajul de programare, dar și cu situri despre șarpele cu același nume.

**reducerea
ambigui-
tății**

Ce-i de făcut? Căutăm pe Google folosind tehnica exemplificată de secvența: **python -snake -monty**. Semnul minus pus în fața unor

⁹⁹**wget** este un program din lumea Unix, unde acesta este modul de a opri forțat rularea unui program. Programele MS-DOS sunt oprite, probabil, de **CTRL+BREAK**. În Win98 apăsați **CTRL+ALT+DEL** și opriți procesul respectiv.

¹⁰⁰Din practică știu că se întâmplă ca o persoană să aibă în pagina sa texte pe care aș vrea să le citesc și care nu ocupă mult loc pe disc, dar și fișiere enorme cu muzică sau filme, de care n-am nevoie și care ar putea epuiza spațiul de pe disc. Sub sistemele Unix sau WindowsNT, în mod normal, utilizatorii obișnuiți au alocată doar o porțiune limitată de disc. S-ar putea ca un film să nici să nu încapă pe zona care le-a fost alocată.

¹⁰¹Pentru mai multe detalii despre **wget** și pentru problemele ridicate de combinarea diverselor opțiuni consultați documentația programului.

cuvinte-cheie îi spune motorului de căutare să excludă de pe lista pe care o produce siturile despre șerpi sau „monty“.

Persoanele care sunt în căutarea șarpelui și nu a limbajului de programare pot să pună `python -programming`.

O altă restrângere a căutărilor rezultă evident atunci când ne limită la situri care sunt numai într-o anumită limbă. Acest lucru se poate face explicit sau implicit: cuvintele folosite sunt, să zicem, în limba română. 5

În afara limbii, o altă posibilitate de a restrânge aria siturilor investigate o reprezintă folosirea unui motor de căutare specializat. Putem porni însă de la un motor de căutare general și de aici să descoperim siturile cu motoare de căutare specializate. Google are chiar liste gata făcute cu asemenea situri specializate. 10

Dacă motorul de căutare permite acest lucru, putem desigur aplica și tehnicile mai subtile ale expresiilor regulate.¹⁰² 15

A.5 Corectura computerizată

Există multe metode și programe de corectare, în special pentru limba engleză. Noi vom prezenta aici posibilitatea de a efectua o corectură ortografică în mediul integrat TeXnicCenter.

TeXnicCenter folosește aceeași mașinărie pentru corectură ca și **MySpell** OpenOffice. Numele tehnic al bibliotecii respective de funcții este „MySpell“¹⁰³. 20

Pe situl <www.openoffice.org> puteți găsi legăturile necesare pentru a descărca o sumedenie de dicționare ortografice.

Soluția pe care o descriem aici este una care poate fi dezvoltată independent de existența unui dicționar gata făcut. În practică, am avut nevoie de ea pentru că mai demult nu exista pe situl OpenOffice un dicționar ortografic pentru limba română.¹⁰⁴ 25

¹⁰² Dacă n-ați examinat încă anexa despre expresiile regulate, puteți să o faceți acum; vezi §A.3.

¹⁰³ Autorul MySpell este Kevin B. Hendricks. Algoritmii utilizați pentru a opera cu afixe sunt bazați pe cei ai lui Geoff Kuenning, autorul programului Ispell. Persoanele interesate de detalii pot citi cu folos sursele programului TeXnicCenter în partea lor care privește MySpell.

¹⁰⁴ La data de 22/10/2003 am găsit un dicționar MySpell creat de Nicu Buculei pornind de la dicționarul Ispell al lui Mihai Budiu <<http://www.cs.cmu.edu/~mihai>>. Este un dicționar destul de amplu, dar fără semnele diacritice din dicționarul lui Budiu. RomanianOffice, care este un program de birou bazat pe OpenOffice, include un dicționar cu diacritice, dar dicționarul este proprietatea firmei INTERSOL. Chiar dacă versiunea 1 a produsului RomanianOffice este

L-am improvisat în felul descris pe scurt în continuare. Primul pas constă în construirea unui fișier `ro_FI.dic` pe care trebuie să-l plasați în dosarul **Language** aflat pe calea folosită la instalarea **T_EXnicCenter**. În fișierul acesta plasați o listă de cuvinte, corect
5 ortografiate, puse fiecare la începutul unui rând și sortate alfabetic. Primul rând al acestui fișier este rezervat pentru numărul de rânduri din fișier. Nu este greu să stabiliți, cu ajutorul Vim, numărul de rânduri.

Mai trebuie să creați un fișier `ro_FI.aff` pe care-l puneți tot
10 în dosarul **Language** din dosarul unde este instalat **T_EXnicCenter**. Rațiunea existenței acestui fișier este foarte simplă. O voi explica cu ajutorul unui mic fragment dintr-un asemenea fișier:

```
1 SET ISO8859-2
2
3 PFX G Y 1
4 PFX G 0      meta
```

Primul rând specifică pentru corectorul ortografic codificarea folosită. Rândul al patrulea îi spune sistemului cum să atașeze un prefix,
15 foarte important pentru filosofie și nu numai, prefixul *meta*. Se observă că aici a fost declarat un *steag*, cum se spune tehnic.¹⁰⁵ Steagul trebuie atașat cuvintelor din fișierul de tip `dic` care pot fi prefixate cu *meta*. Iată două exemple evidente:

```
1 filosofie/G
2 limbaj/G
```

Sistemul va recunoaște acum drept corecte și cuvintele „metafilosofie” și „metalimbaj”.
20

Nu rămâne de făcut decât un mic reglaj în **T_EXnicCenter**. În meniul **Tools** mergeți la rubrica **Options...** și apoi la panoul **Spelling**. Alegeți `ro` în caseta **Language**. Observați cum apare automat **FI** („româna Filosofică”) în caseta **Dialect**. Nu rămâne decât să bifați
25 **Check spelling while typing** și sistemul va colora cuvintele pe care nu le găsește în listă sau nu le poate obține din cuvintele din listă folosind metodele descrise în fișierul de tip `aff`.

În versiunile mai vechi ale **T_EXnicCenter** nu se putea folosi decât **Spelling...** din meniul **Tools**. Începând cu versiunea 1 Beta 6.20

gratuită, nu puteți folosi dicționarul decât în cadrul acestui produs-program. În consecință, cred că este în continuare interesant de lucrat la un dicționar plasat sub o licență de tip GPL.

¹⁰⁵ În limba engleză, termenul este *flag*.

A. Creionul electronic

Beta, este posibil să dăm un clic pe butonul din dreapta al *mouse*-ului, pe cuvântul marcat ca incorect și să folosim un meniu contextual.

Anexa B

Tehnoredactarea computerizată

B.1 L^AT_EX

- 5 Programul de care ne-am folosit pentru a tehnoredacta cartea de față s-a născut din nevoi pur practice. Autorul său, Donald E. Knuth, era absorbit de munca la tratatul său despre *Arta programării calculatoarelor*¹. Lucrul la tratat l-a început în 1962, pe vremea când cărțile se tipăreau în mod tradițional. După 1970, în SUA, a
- 10 început trecerea la producerea cărților cu ajutorul sistemelor computerizate. Donald Knuth a fost însă îngrozit de forma pe care o avea tiparul electronic la începuturile sale. El s-a decis să realizeze propriul său program de creat cărți frumoase. Generos, el a plasat sistemul său zis T_EX în domeniul public.²

**Donald
Knuth**

- 15 Procesul prin care un text ajungea, în veacul trecut, să fie tipărit ar putea fi segmentat în următoarele faze: autorul crea un manuscris; manuscrisul era dactilografiat; textul dactilografiat era prelucrat într-o redacție; după redactare, textul era tehnoredactat; după ce era tehnoredactat se dădea la cules, adică se crea pentru

¹Vezi Donald E. Knuth, *Arta programării calculatoarelor*, 3 volume (București: Teora, 1999-2002). Aceasta este o carte celebră. Așa cum reiese din textul reprodus pe ultima copertă a tuturor celor trei volume, Bill Gates a scris despre ea următoarele: „Dacă te crezi programator, ... citește *Arta programării calculatoarelor* de Knuth. ... Dacă poți citi toată cartea, trimite-mi neapărat un C.V.“ Trebuie să țineți cont că aceste cuvinte vin nu doar din partea unuia dintre cei mai mari oameni de afaceri din toate timpurile, ci și a cuiva care a publicat ca programator-cercetător (a se vedea B. Gates și C. Papadimitriou, „Bounds for sorting by prefix reversals“, *Discrete Mathematics* 27: 47-57, 1979)

²O excelentă prezentare a bazelor sistemului T_EX, în limba română, o găsiți în cartea lui Moroșanu[30].

fiecare pagină o pagină din litere de plumb; textul cules era dat la corectură; varianta finală se tipărea folosind paginile de plumb.

Faza tehnoredactării însemna punerea pe manuscris a fel și fel de semne pe care tipografiile le citeau pentru a ști cum să meștească paginile lor de plumb. Pe de o parte, trebuie să identificați existența unui limbaj al tehnoredactării. Pe de altă parte, trebuie să țineți cont că realizarea formelor în care erau turnate literele de plumb era o artă.³

Donald Knuth a fost extrem de dezamăgit de decăderea bruscă a artei tiparului sub impactul computerizării. El a rezistat tentației de se crampona de vechile litere de plumb și a folosit fantasticele sale calități de programator pentru a realiza un sistem de așezare a textului în pagină probabil fără egal.

B.1.1 Utilizarea programului \LaTeX : costuri și beneficii

Ce este \LaTeX ? Fără a intra în detalii, putem spune că \LaTeX vă permite să tipăriți (pe hârtie sau în format electronic), cu ajutorul în ultimă instanță al \TeX , eseurile dumneavoastră. Aceleași tehnici pot fi folosite pentru a tipări practic orice: articole, cărți, afișe și așa mai departe. \LaTeX a fost creat de către Leslie Lamport.⁴

Leslie
Lamport

Cât este de greu de folosit sistemul \LaTeX ? Nu vă grăbiți cu această întrebare! Mai bine faceți un calcul cu privire la costuri și beneficii.

De ce să nu folosim doar Vim pentru a da forma finală eseurilor? Vim este precum un stilou sau o mașină de scris. \LaTeX este ca un tipograf.

Vestea bună ar fi că puteți beneficia de posibilitățile unei edituri și ale unei tipografii, folosind \LaTeX și programele asociate.

Și vestea proastă? Este chiar proastă dacă aveți nici un fel de cunoștințe despre programarea calculatoarelor. \LaTeX , ca și \TeX , este un limbaj de programare. Pentru a tehnoredacta eseuul și a-l tipări frumos trebuie să scrieți un program.⁵

Care sunt acum rezultatele analizei cost-beneficii? Cred că sunt

³Istoria acestei arte este povestită de S.Tóth[45].

⁴Prezentarea standard a sistemului este cea pe care o face chiar Leslie Lamport[21]. Cartea cuprinde un manual de utilizare și o descriere sistematică a \LaTeX . În limba română, există o prezentare a versiunii mai vechi a \LaTeX (Pusztai și Ardelean[38]). Blaga și Pop[4] prezintă $\text{\LaTeX}2\epsilon$.

⁵Moroșanu[30, p.10] subliniază foarte limpede caracterul de limbaj de programare al \TeX .

în favoarea \LaTeX . Dacă scrieți un eseu filosofic, nivelul pe care trebuie să-l atingeți în domeniul programării în \LaTeX este cât se poate de elementar. Practic, vă trebuie doar câteva zile ca să învățați chiar dacă n-ați programat niciodată. Costul acesta (care nu este nul)⁶ este contrabalansat din plin de calitatea rezultatului obținut.

**programe
elementară**

La beneficiile unui text frumos tipărit se adaugă însă și disciplina intelectuală pe care o impune utilizarea \LaTeX . Acesta este un câștig mai puțin vizibil, dar extrem de consistent. \TeX , \LaTeX , \BibTeX și programele asociate sunt realizate de minți strălucitoare din mediul academic. Dacă elaborați eseuri în mediul universitar, atunci spiritul acestor programe este cel de care aveți nevoie.

**disciplina
intelectuală**

B.1.1.1 Avantajul de a face față complexității rescrierii textului

În aparență obiectivul este modest: așezarea textului în pagină. În cazul Vim am fost preocupați de crearea de fișiere care conțin text, de căutarea de șiruri de semne într-un text și de substituirea lor cu alte semne. Acum problema este de a aranja textul în pagină. Pentru a așeza textul în pagină trebuie să ne angajăm însă deplin pe calea programării. Ca și vechii tehnoredactori care-și scriau pe manuscris mesajele către tipografi, noii tehnoredactori trebuie să trimită mesaje unui program care va aranja textul.

Mesajele către programele care vor prelucra textul pe computer sunt scrise într-un limbaj. Chiar dacă uneori pot să nu pară prea complicate, în spatele lor se află algoritmi, specificații ale pașilor care trebuie făcuți pentru a rezolva o problemă.

În ciuda obiectivului principal aparent modest, Knuth a pus bazele unui sistem care face o mulțime de lucruri. Între altele, sistemul poate gestiona structura unui articol sau a unei cărți. Poate ține evidența și plasa automat în pagină notele de subsol sau notele marginale. Inserează imagini. Creează desene, scheme de care au nevoie autorii. Realizează indici. Ține evidența listei bibliografice. Asigură posibilitatea de a face trimiteri interne sau la documente aflate pe Internet.

**capacitățile
 \TeX**

Ar fi greu de descris tot ce poate face o distribuție \TeX . Conceput inițial pentru știința computerelor, sistemul a fost adoptat de către matematicieni. Din punctul de vedere al filosofiei, aceasta înseamnă

⁶Cu un procesor de cuvinte puteți scrie un text din prima clipă. Problemele vin abia când vreți să faceți lucruri mai complicate. În cazul \LaTeX lucrurile stau exact invers: este mai greu la început, dar este mai ușor ulterior.

că există în el o sumedenie de mijloace pentru a tehnoredacta texte de logică simbolică. Este însă cât se poate de util și dacă vrem să scriem în greaca veche.

compilare S-ar putea ca vestea proastă pentru mulți să fie aceea că, pentru a lucra în \TeX , trebuie să scrii o sursă care va fi supusă procesului de compilare. Programele obișnuite de birou tehnoredactează totul din mers, interpretând fiecare pas făcut de utilizator. Procesul de compilare este diferit. Sursa este prelucrată în întregime de ea. Erorile de programare pot duce la oprirea procesului de compilare ca atare. Procesul de realizare a produsului final este mai dificil, dar rezultatul obținut are altă calitate. 5 10

Un sistem \TeX , după cum se vede din cele scrise mai sus, este altceva decât o suită de programe pentru munca de birou.

rescrierea textului Marea calitate a unui sistem \TeX este că permite soluționarea problemelor legate de rescrierea textului. Dacă n-ar trebui să rescriem un text, atunci sistemele uzuale ar fi destul de potrivite. În realitate, munca intelectuală presupune așternerea unor straturi succesive ale textului. Aici se vede, dacă treceti peste dificultățile începutului, superioritatea sistemului lui Donald Knuth. 15

Evident, frumusețea aranjării finale în pagină nu este de neglijat nici ea. Dar, în practică, chiar dacă editorul m-a forțat să-i dau textul în formatul fișierelor unui editor de birou, tot a fost mai ușor să scriu în sistemul lui Knuth.⁷ 20

B.1.1.2 Avantajele pentru cine vrea să aibă o perspectivă generală asupra științei computerelor 25

Sunt toate acestea doar simple unelte? Punctul de plecare al sistemului a fost de ordin practic, dar el a luat turnura necesară pentru a fi punctul de pornire și al unor reflecții cu caracter general. O anexă la un ghid de elaborare a eseurilor filosofice nu oferă cadrul necesar pentru a duce prea departe considerațiile teoretice. Putem face doar scurte observații despre natura algoritmilor și a proceselor algoritmice. 30

⁷Cel care m-a convins definitiv de superioritatea sistemului lui Knuth este cunoscutul logician Melvin Fitting. El mi-a explicat că, într-adevăr, își scrie scrisorile cu un editor de birou; cărțile și articolele sale de logică sunt scrise însă cu un sistem \TeX . Distincția este limpede și convingătoare: dacă scrisorile ar fi rodul unui proiect de cercetare, dac-ar avea o structură complicată, numeroase trimiteri etc., atunci ar trebui și ele scrise cu un sistem gen \TeX . O scrisoare are, de multe ori, o singură pagină și este mult mai comod să aranjezi direct, vizual elementele ei în pagină decât să scrii un program. O carte are sute de pagini. Aici avantajul programării se face repede simțit.

**probă de
literă vs.
literă**

Începem cu o distincție care amintește de distincția dintre tip și mostră. Dacă aranjăm în pagină șirul de semne validitate, câte li-

tere „a“, „i“ sau „t“ trebuie să plasăm? Una sau două? Putem formula
răspunsul în termenii distincției tip-mostră (sau probă de literă)⁸.

5 În „validitate“ există șapte tipuri de literă, iar din tipurile „a“, „i“ și „t“ avem câte două mostre sau probe de literă.

În sistemul lui Knuth fiecare literă are un cod. La rândul lor, literele sunt grupate pe categorii. O probă, în sistemul lui Knuth, este desemnată printr-o pereche formată dintr-un cod și o categorie.

10 Operarea cu o probă seamănă cu felul în care, în vechile tipografii, zețarii⁹ culegeau cu mâna literele din niște cutii. Aveau cutii cu litere de unde luau ce le trebuia și așezau în pagină. Aceasta nu este altceva decât esența tiparului lui Gutenberg: literele mobile. Pe de
altă parte, un „a“ poate avea categorii diferite. Așa cum, în Vim,
15 într-un mod este comanda de a plasa litera a în text, iar în alt mod o comandă de trecere la inserarea de text după poziția curentă a cursorului.

Mai departe, putem rafina ideea de procesor. \TeX are un număr limitat de comenzi de bază. Aceste comenzi pot fi folosite pentru a
20 scrie alte comenzi mai complexe. Cu alte cuvinte, în inima sistemului conceput de Donald Knuth există un procesor virtual.

Când vorbesc despre procesoare oamenii se gândesc la procesoarele fizice. Putem face însă abstracție de ele și să ne concentrăm
asupra a ceea ce face procesorul virtual.

procesor

25 În sistemul lui Donald Knuth, putem face o distincție între două procese fundamentale: plasarea unor cutii virtuale pe pagină și ceea ce este pus în cutiile respective. Sistemul nu plasează deci literele ca atare, ci cutiile în care sunt puse probe de litere. Putem astfel separa procesul de aranjare a cutiilor de procesul de desenare a literelor.

30 Aici ne vom concentra atenția, în primul rând, asupra aranjării cutiilor și alegerii unui conținut potrivit pentru ele. La urma urmei, nu refacem desenul tipurilor de literă de fiecare dată când scriem un text. Folosim probele de litere existente și variantele lor pentru a da un conținut cutiilor. Limbajul care ne permite acest lucru ne
35 interesează cu precădere.

Ca și-n cazul procesoarelor reale, nu este tocmai ușor să programăm direct în limbajul procesorului virtual. În multe situații putem

⁸Tipografii colecționează exemple din literele de care dispun și alcătuiesc albume care se numesc „probare de litere“ (v. *Dicționarul limbii române* (București: Editura Academiei, 1984), tom VIII, partea a 5-a, s.v.).

⁹În limba veche se numeau chiar „probari“.

face abstracție de ceea ce se întâmplă la nivelul procesorului. Ca și atunci când folosim limbaje de nivel mai înalt, și-n cazul sistemului lui Donald Knuth putem utiliza sisteme de comenzi mai complexe, care vor fi traduse automat în procesul compilării.

Compilarea înseamnă în sistemul lui Knuth generarea a unor fișiere care pot fi vizualizate și tipărite. Pentru vizualizare și tipărire vor fi folosite programe precum Ghostview sau Acrobat Reader.

limbaje de
nivel înalt

Sistemul de comenzi complexe de nivel mai înalt pe care-l vom folosi este \LaTeX . Ca și limbajele de nivel înalt, el facilitează enorm programarea. De asemenea, ne ajută să nu „reinventăm roata”. Dacă avem o problemă și suntem începători, atunci este aproape sigur că altcineva a găsit o soluție. \LaTeX ne permite să integrăm comod asemenea soluții în propriul nostru program.

B.1.2 Instalarea \TeX și a programelor asociate

Distribuția \TeX la care ne referim constant în această carte se numește Mik \TeX . Puteți găsi instrucțiunile de instalare la adresa <http://www.miktex.org>. Primul pas este obținerea unei colecții de arhive cu ajutorul unui program de descărcare-instalare. Presupun că veți descărca mai întâi arhivele pe propriul computer. Cum volumul Mik \TeX este foarte mare, s-ar putea să fie mai rezonabil să obțineți de la laboratorul Facultății un CD cu fișierele Mik \TeX .

Asigurați-vă apoi c-ați instalat Ghostscript, Ghostview și Adobe Acrobat.

Instalarea ca atare nu presupune nimic deosebit. Este suficient să urmați instrucțiunile. Recomandarea mea ar fi să nu economisiți spațiul de pe disc și să instalați sistemul \TeX în forma sa completă. Puneți, de asemenea, sistemul \TeX într-un dosar special și pe o cale care nu conține nume de dosare cu spații. Ați putea folosi, de pildă, ceva de genul `D:\ed\tex`. Puneți în dosarul `tex` atât dosarul `texmf`, dosarul principal al distribuției Mik \TeX , cât și `localtexmf`. Tot în `tex` aș sugera să fie pus tot ce este legat de \TeX .

Instalarea totală a distribuției Mik \TeX poate să ia destul de mult timp.¹⁰ În momentul când scriu aceste rânduri, ultima versiune a Mik \TeX , versiunea 2.4, instalează 44850 de fișiere (929 de pachete). Pe computerul cu care este scrisă această carte dosarul `tex`, cu toate

¹⁰Se consumă timp și cu verificarea integrității pachetelor. Acest proces este însă absolut necesar. S-ar putea ca unele fișiere să se fi descărcat parțial de pe Internet. De asemenea, procesul acesta de verificare vă ferește de nedorite coruperi ale fișierelor pe parcursul circuitului lor pe Internet sau al scrierii pe CD.

programele asociate, ocupă 654MB și conține 46758 de fișiere în 4566 dosare.

Dacă n-aveți suficient spațiu pe discul dur, instalați doar o versiune redusă a distribuției. Aveți o opțiune în acest sens chiar în programul de instalare. Mai puteți elimina, de asemenea, din documentația sau pachetele pe care nu le folosiți foarte des. Folosiți pentru aceasta programul MikT_EX Package Manager.

Configurarea sistemului MikT_EX nu este deosebit de complicată. Asigurați-vă de existența în AUTOEXEC.BAT, după SET PATH, a unei căi de genul D:\ED\TEX\TEXMF\MIKTEX\BIN. Aceasta este calea din Windows unde sunt puse executabilele MikT_EX. Dacă ea nu există, adăugați calea adecvată sistemului dumneavoastră. Nu uitați că trebuie să separați căile prin punct și virgulă (vedeți modelul din subsecțiunea A.1.4.4).

N-ar fi rău să reorganizați și Start Menu în așa fel încât să nu fie toate scurtăturile puse în Programs. În orice caz, găsiți scurtătura care se numește MikTeX Options și care trimite către mo.exe, unul dintre executabilele MikT_EX. Executați un clic pe Languages și bifați limbile de care aveți nevoie. În orice caz, bifați romanian. Dați apoi clic pe General și, pentru siguranță, un clic pe Refresh Now. Această împrăștiere a bazei de date este absolut necesară când adăugați noi pachete cu fișiere.

Sistemul este efectiv uriaș și integrează o contribuții realizate de către diverși autori. Porniți MikTeX Package Manager și veți avea o listă a pachetelor instalate, cu unele scurte informații despre ceea ce face fiecare.¹¹

După configurare, Win98 trebuie repornit. Dacă totul a mers bine, ar trebui ca sistemul MikT_EX să fie funcțional.

B.1.2.1 Sub ce sisteme de operare funcționează T_EX?

Întrebarea din titlu este foarte importantă. Dacă intrați în sala calculatoarelor de la Facultatea de Filosofie, observați imediat computerele Sun și PC-urile care funcționează sub sistemul de operare Linux. Sistemele de operare sunt, în acest caz, de tip Unix. Alături de ele veți găsi însă cel puțin un PC care folosește Windows.

Multe utilizatoare și utilizatori de sisteme de calcul au acasă sau la serviciu WindowsXP, nu Windows98. Merge L^AT_EX sub toate aceste sisteme de operare? Răspunsul este categoric da.

¹¹ Acest program există începând cu versiunea 2.3 a MikT_EX, dar lipsește din versiunile anterioare.

Instalarea sub WindowsXP este asemănătoare cu aceea de sub Windows98. Atenție doar la setarea specifică a variabilei de mediu pentru calea pe care se află executabilele MikTeX.¹² Din experiența noastră concretă am putea spune că MikTeX se instalează mai lesne sub WindowsXP. Într-unul dintre cazuri instalarea a fost foarte dificilă, pe același computer, sub Windows98 și lină sub WindowsXP. Sub Windows98, computerul s-a blocat de câteva ori. WindowsXP gospodărește evident mai bine resursele calculatorului și asigură o funcționare mai stabilă.

Dar Linux? L^AT_EX este la el acasă sub Linux. Orice distribuție Linux importantă include T_EX și programele asociate. Numele distribuției respective, sub Linux, este TeT_EX.

N-am o experiență directă de utilizator de L^AT_EX sub alte sisteme Unix. Din câte știu, lucrurile ar trebui să decurgă fără probleme. Aceași este situația și-n cazul Mac.

B.1.2.2 Ce se întâmplă dacă n-aveți computer sau ai unul foarte vechi?

Aș recomanda însă L^AT_EX studentelor și studenților care nu au acasă un computer sau care au un model foarte vechi.¹³ Ce poți face cu un computer foarte vechi? Poți scrie fișiere de tip text! Poți folosi o versiune mai veche a Vim, eventual fără interfață grafică.

Pentru sursele L^AT_EX nu este nevoie decât de fișiere de tip text. O carte întreagă încapă fără probleme pe o singură dischetă. Sursele respective pot fi apoi compilate și corectate sub Linux, într-un laborator care dispune de computere performante.

Puteți folosi pentru a crea surse L^AT_EX chiar și un calculator care n-are disc dur! Un PC foarte vechi, de la începutul anilor '80 ai secolului trecut, cu doar două dischete poate fi folosit cu mult succes. Un calculator de tip Spectrum, dar care are și sistemul de operare CP/M (cum este calculatorul românesc HC2000), poate servi și el la producerea fișierelor text care alcătuiesc sursa L^AT_EX.

Nu uitați că L^AT_EX sau programele asociate pot produce și desene pornind de la fișiere de tip text!

Achiziționarea de sisteme scumpe doar pentru a scrie eseuri pentru examene, lucrări de diplomă, articole, teze de doctorat, cărți

¹²Vezi aici explicațiile de la pagina 140, rândul 8.

¹³Astăzi, un model vechi de computer pe care să meargă bătrânul MS-DOS sau o versiune veche de Windows poate fi achiziționat la un preț care oscilează între salariul minim și cel mediu din România. Un sistem foarte vechi are un preț chiar mai mic.

este efectiv o eroare. Investiția trebuie făcută în direcția învățării programării, nu a utilizării de computere foarte scumpe.

Singurul lucru de care este nevoie, pe lângă computerul ieftin care produce fișierele de tip text, este accesul la un laborator cu
5 computere pe care este instalat L^AT_EX.

La limită, puteți sta și scrie chiar în laborator (sau la un Internet café). Rezultatul depinde de mintea celei sau celui care scrie, nu de prețul plătit pe calculator.¹⁴

B.1.2.3 Mediul integrat de dezvoltare T_EXnicCenter

10 În configurația pe care o aveți în acest moment trebuie să folosiți comenzile în linie într-o fereastră MS-DOS pentru a compila sursele L^AT_EX. Aș sugera că este absolut utilă instalarea unui mediu integrat de dezvoltare de programe L^AT_EX.¹⁵

Există și un script Vim, creat de Fritz Mehner, care creează me-
15 niuri, inclusiv pentru compilare și vizualizare.¹⁶ Scriptul trebuie configurat pentru a lucra adecvat.¹⁷ S-ar putea să doriți să folosiți acest script sub Linux pentru a avea acces rapid la compilare și vizualizare.

Am arătat deja cum putem găsi pe Internet locul de unde poate
20 fi descărcat T_EXnicCenter.¹⁸ Recomandarea ar fi să instalați acest program într-un dosar plasat în dosarul în care aveți și MikT_EX.

Instalarea ca atare n-ar trebui să vă creeze probleme, dacă urmați instrucțiunile. Versiunea care a fost folosită pentru a compila sursele cărții de față are însă o particularitate care s-ar putea să vă
25 atragă atenția; se numește „1 Beta 6.01”. Ce semnificație are „Beta”? Programele sunt texte și lor li se aplică din plin principiul rescrierii. Ele sunt rescrise până se ajunge la o versiune matură. O versiune beta este încă într-un stadiu în care mai sunt erori. Este însă su-

¹⁴În practică, se întâmplă, uneori, să primești eseuri scrise în limba română, dar fără diacritice! Ce ați zice dacă vi se arată, prin contrast, un text cu diacritice și citate frumos realizate, în greaca veche?

¹⁵Este vorba aici mai ales de utilizatorii sistemului de operare Windows. Linux transformă computerul într-o puternică stație de lucru. Cine este capabilă sau capabil să lucreze în Linux se poate descurca folosind Makefile-uri. De asemenea, comanda în linie din Linux este incomparabilă ca putere cu ceea ce oferă sistemul Windows ca atare.

¹⁶Vezi <<http://lug.mfh-iserlohn.de/vim/vim-latex/vim-latex.html>> pentru explicații, exemple de utilizare și descărcarea scriptului lui Mehner.

¹⁷N-am testat scriptul sub Linux, dar acesta pare conceput sub un sistem Unix.

¹⁸A se vedea aici pagina 177, rândul 27.

ficient de bună pentru a fi utilizată, iar autorii aşteaptă mesajele utilizatorilor cu privire la eventuale deficienţe.

În ciuda celui „beta” din versiunea folosită de către noi, mediul integrat funcţionează destul de bine. Este totuşi recomandat să-l folosiţi exclusiv pentru a compila sursele.¹⁹ Pentru scrierea surselor ar fi mult mai sigur să folosiţi Vim.

Configurarea \TeX nicCenter este ceva mai problematică decât instalarea. Când porneşte pentru prima oară \TeX nicCenter lansează automat un Wizard²⁰ care configurează mediul integrat.

Dacă mergeţi pe ruta **Build**→ **Define Output Profile...** şi daţi un clic pe butonul **Wizard**, puteţi porni oricând doriţi programul de configurare.

Dacă pe computer este instalat editorul Acrobat, editorul de fişiere pdf al firmei Adobe, atunci s-ar putea să obţineţi automat o configurare în care fişierele de tip ps (fişierele PostScript) sunt deschise de către programul **distiller**, care converteşte un fişier de tip ps într-unul pdf. Dacă doriţi doar să vizualizaţi fişierul PostScript, trebuie să modificaţi manual configuraţia \TeX nicCenter sau să faceţi în aşa fel încât Ghostview să fie programul care deschide automat fişierele pdf sub Windows.

Ce trebuie să ştiţi pentru a configura mediul integrat? În primul rând, trebuie să ştiţi unde sunt plasate executabilele Mik \TeX ²¹ Căile către Ghostscript şi Acrobat Reader ar trebui să fie identificate automat. Evident, este bine însă să le cunoaşteţi dinainte.

Singurele probleme serioase cu configurarea automată, sub Windows98, le-am avut în cazul Acrobat Reader. Dacă vedeţi, pe parcurs, că Acrobat Reader nu porneşte când îl chemaţi din mediul integrat, mergeţi pe ruta **Build**→**Define Output Profile...**, selectaţi **LaTeX**⇒**PDF** şi daţi clic pe **Viewer**. Veţi vedea acolo trei comenzi. Dacă este o comandă DDE, la primele două puneţi **[FileOpen ("%bm.pdf")]**. Acum ar trebui să puteţi deschide Reader-ul. Alternativ, alegeţi comanda în linie şi puneţi **"%bm.pdf"** la primele două comenzi. Problema care rămâne este legată de rescrierea fişierului pdf deschis în Reader. Reader-ul nu permite modificarea fişierului

¹⁹Modul acesta de lucru este, de altfel, imperios necesar dacă n-aveţi la dispoziţie în mod curent un calculator pe care este instalat \LaTeX . Creaţi sursele programelor separat şi apoi executaţi ciclul compilare-vizualizare-corectare până obţineţi rezultatul dorit.

²⁰Cuvântul acesta înseamnă în engleză *wizard*. n-are însă sens să-l traducem. Aici este folosit ca o simplă etichetă.

²¹Pentru conceptul de „executabile Mik \TeX ” se vedea aici explicaţia de la pagina 189.

pe care l-a deschis. Acesta ar trebui închis. Dacă n-o face mediul integrat, închideți manual fișierul direct în Reader.

Sub WindowsXP, T_EXnicCenter a configurat în mod automat vizualizarea cu Acrobat Reader în modul descris mai sus. Posibilitatea de a închide automat fișierul pdf nu există. Dacă nu-l închideți, veți primi un mesaj de eroare de la compilator.

Versiunea mai nouă, 1 Beta 6.20, integrează mai bine Reader-ul. Ea are și un corector ortografic mai funcțional, precum și o serie de alte îmbunătățiri.

B.1.2.3.1 Integrarea Vim în T_EXnicCenter T_EXnicCenter are propriul său editor de texte. Are, de asemenea, meniuri care permit introducerea direct în mediul integrat a comenzilor din limbajul L^AT_EX.

Este posibil însă să integrați editorul Vim în T_EXnicCenter. Mergeți la meniul Tools→Customize și dați un clic pe Tools, iar apoi un clic pe pictograma New. Dați un clic pe butonul din dreptul casetei Command și procedați ca și cum ați deschide fișierul gvim.exe (principalul executabil din dosarul unde este instalat Vim). La argumentele comenzii puneți +%1 %pc.²² La directorul inițial puneți %dc. Ați indicat astfel că vreți să deschideți documentul curent din T_EXnicCenter în Vim.

**deschideți
Vim la
rândul
curent**

Vim și T_EXnicCenter colaborează bine și, dacă ați modificat textul în Vim, T_EXnicCenter vă va întreba dacă să modifice la rândul său textul. Invers, dacă ați modificat textul în T_EXnicCenter, la revenirea în Vim dați clic pe Load File în caseta de dialog care va apărea automat. Preluati astfel automat schimbările. Altfel, le pierdeți! Veți primi, ce-i drept, niște mesaje din partea Vim.

Pentru a verifica dacă totul merge fără probleme este imperios necesar să exersați pe fișiere test. Nu treceți direct la scrierea eseului pe care trebuie să-l predați mâine. S-ar putea să aveți surprize neplăcute.

Multe cititoare și cititori se vor fi întrebând însă ce rost are să folosim Vim. Mediul integrat are o mulțime de pictograme care ne permit să scriem comenzile L^AT_EX fără efort. Limbajul L^AT_EX este extrem de subtil. Ar fi imposibil pentru cineva să creeze exact comenzile de care aveți nevoie. Recomandarea noastră este să creați propriile dumneavoastră meniuri. Alternativ, dacă n-aveți interfață grafică sau nu vă plac meniurile, folosiți scripturi Vim. Puteți crea

**creați
meniuri
proprii**

²²Litera 1 vine de la *line*; nu o confundați cu cifra 1. Studiați opțiunile la invocarea în linia de comandă a lui Vim cu ajutorul comenzii gvim.exe -help.

ușor scripturi Vim folosind doar al treilea bloc de cod din exemplele noastre cu meniuri Vim.²³

La urmă, dar nu în cele din urmă, cred că trebuie accentuată recomandarea de a folosi separat Vim de mediul integrat cât timp construim sursele. Cu alte cuvinte, editorul Vim (sau unul similar) trebuie folosit pentru a introduce textul ca atare. Folosim mediul integrat doar pentru a compila sursele și pentru a le corecta. În acel moment s-ar putea să avem nevoie de funcționarea integrată și a editorului Vim.²⁴

B.1.2.3.1.1 Vim și programul de vizualizare a fișierelor dvi După compilarea sursei \LaTeX rezultă un fișier dvi care este vizualizat cu ajutorul programului Yap din distribuția \TeX . La instalarea \TeX nicCenter, Yap este setat în așa fel încât un dublu clic ne readuce în editorul mediului integrat.

Mergeți în Yap pe ruta View→ Options... În panoul Inverse Search modificați comanda după modelul următor:

```
D:\use\Vim\vim62\gvim.EXE +%l %f
```

Fiți atente și atenți să nu repuneți în drepturi \TeX nicCenter: nu selectați nimic în caseta pentru programe! Secretul comenzii către Vim este +%l care-i spune să deschidă fișierul-sursă la linia 1. Puneți evident calea potrivită pentru felul în care este instalat Vim.

S-ar putea să fie necesar să verificați și registrul Windows, dacă nu merge totul cum trebuie. Cu toate că pare mai dificil de operat, schimbarea aceasta este foarte utilă când vreți să operați corecturi mai complicate în fișier.

²³Cum porniți un script Vim? Tastați, în modul normal, comanda :source sau, mai scurt, :so și apoi calea unde se găsește scriptul, respectiv doar numele scriptului vim, când acesta se află în dosarul curent. De asemenea, istoria comenzilor în linie este de mare ajutor. Căutați și un fișier _viminfo. S-ar putea să fie în home, dacă aveți așa ceva. Acolo găsiți istoria comenzilor. Ce vi se pare mai reușit puteți păstra în alt fișier în vederea construirii de script-uri Vim.

²⁴Recomandarea aceasta este făcută și pentru că mediul integrat recomandat aici este într-o versiune beta. N-am avut accidente majore cu editorul său intern. Impresia mea este totuși că nu manevrează bine fișiere mari. Vim este mult mai sigur și mai flexibil.

B.1.3 L^AT_EX într-o săptămână

L^AT_EX este un limbaj de programare. Trebuie învățat ca orice limbaj de programare: făcând exerciții. Exercițiile propuse de către noi aici constau în construirea unor meniuri Vim foarte simple.²⁵

- 5 De data aceasta meniurile Vim nu sunt un scop în sine, ci un mijloc de a tasta mai rapid componentele diverselor construcții posibile în L^AT_EX. Atenția trebuie să fie concentrată asupra comenzilor L^AT_EX ca atare, nu asupra modului în care sunt definite meniurile în limbajul Vim.
- 10 Independent de meniuri, comenzile L^AT_EX pot fi introduse cu ajutorul oricărui editor de texte prin simpla tastare a textului comenzilor.

B.1.3.1 Prima zi

- 15 Prima idee care ar trebui învățată este cea de comandă L^AT_EX ca atare. Scheletul unei comenzi L^AT_EX este următorul: \{ }. Între bara oblică inversă și prima acoladă se pune numele comenzii. Între acolade se pune argumentul comenzii. Argumentul nu este altceva decât materialul pe care comanda îl prelucrează în conformitate cu algoritmul aflat în spatele ei.

**schema
comenzi-
lor
L^AT_EX**

- 20 Unde sunt plasate comenzile? În textul eseului ca atare! Ele sunt aïdoma însemnărilor pe care le face tehnoredactorul pe manuscris. Aceste însemnări le folosește apoi tipograful pentru a tipări textul.

- 25 Cum de nu se confundă comenzile ca atare cu elemente similare ale textului? Există o serie de semne rezervate exclusiv pentru comenzi. Ați văzut deja trei dintre aceste semne mai sus. Lista completă a semnelor rezervate este următoarea: \$ & # % _ { } ~ ^ și, la urmă, dar nu ultima în ordinea importanței, bara oblică \. În total 10 semne. Dacă vrem să tipărim aceste semne ca atare, atunci trebuie să dăm niște comenzi prin care să cerem acest lucru.

**semne
rezervate**

- 30 Cineva s-ar putea să fie nedumerit. Ce face L^AT_EX dacă întâlnește semnul, să zicem, s (o literă uzuală din alfabet)? Într-un fel și semnul aceasta este o parte a unei comenzi. L^AT_EX este ca un tipograf. Așa cum tipograful trebuie să știe la ce cutie cu litere din plumb să se ducă, tot așa L^AT_EX trebuie să știe ce literă-ca-tip și ce tip de literă trebuie să tipărească pe ecran sau pe foaia de hârtie. Scepticul s-ar putea să clatine din cap și să mormăie în sinea sa că este prea

²⁵Vezi aici §A.2.2.1.1.

complicat. Dar și-n cazul unui editor uzual de birou trebuie să alegeți tipul de literă!

meniu Exemplul practic s-ar putea să vă convingă însă că este mai simplu decât credeți. Creați un fișier de tip `vim` și introduceți următoarele linii de cod Vim pentru a realiza un meniu:

scheletul

```
unui :imenu ltx1.antet \documentclass[a4paper,11pt]{<Left>
program 2 :imenu ltx1.corp \begin{document}<CR><CR>\end{document}<Up><Home>
LATEX
```

Cele două puncte ne arată că este vorba despre comenzi în linie pentru Vim. Din `imenu` reiese destul de ușor că este vorba despre un meniu activ în modul insert. Numele meniului care apare pe bara principală cu meniuri este `LATEX`. Punctul pus după numele meniului este urmat de numele elementului pe care vrem să-l introducem în meniu.

Dacă nu vreți sau nu puteți lucra în mod grafic cu Vim, experimentați următorul tip de comandă într-un script Vim.

```
1 normal i\documentclass[a4paper,11pt]{}
```

Ideea este aceeași ca și-n primul element al meniului de mai sus. De data aceasta însă, cuvântul-cheie `normal` îi spune lui Vim să treacă din modul comandă în linie în modul comandă normal. Urmează apoi comanda `i`, pentru a insera text, și textul ca atare.

Puteți da comanda de mai sus direct în modul comandă în linie. Aceasta ne arată ce editor formidabil este Vim. Ideile pe care le folosim în construcția interfeței grafice pot fi utilizate și-n absența acesteia, într-un mod cât se poate de eficient.

Ce sens au elementele de meniu? În primul element de meniu, ideea este de a scrie o comandă care să spună despre ce tip de document este vorba. Când dăm clic pe primul element de meniu, Vim scrie în fișier textul cerut și mută cursorul între acolade. Acolo trebuie să scriem despre ce tip de document este vorba. Scrieți, pentru început, `article`. Orice eseu de dimensiuni mai mici este de tipul `article`. De asemenea, pentru a nu vă complica viața de la bun început, considerați că orice eseu mai lung (lucrare de diplomă, disertație, teză de doctorat sau carte) este de tipul `book`. În realitate, `LATEX` este mult mai sofisticat și puteți chiar defini propriile dumneavoastră tipuri de documente.²⁶

²⁶Definirea de tipuri de documente nu este însă o operație pe care o pot face începătorii sau utilizatorii obișnuiți `LATEX`. De altfel, recomandarea ar fi să respectați disciplina impusă de `LATEX` chiar și atunci când știți destul de multe elemente ale limbajului.

Ce sunt parantezele drepte? După cum se vede mai sus, nu fac parte dintre semnele rezervate. În comenzile L^AT_EX se pun între paranteze drepte opțiuni. Evident, L^AT_EX are niște opțiuni standard și ați putea omite cu totul acel element. Noi am vrut să arătăm care este conținutul mai important al acestor opțiuni: formatul foi de hârtie și mărimea literelor. Unitățile de măsură ale literelor se numesc puncte²⁷. Ar fi inutil să facem teoria acestor unități de măsură aici. Experimentând veți vedea ce rezultă pe ecran și pe hârtie.

Opțiunea cea mai importantă pe care s-ar putea să vreți să o controlați este cea a tipăririi pe o singură față a colii de hârtie (*oneside*)²⁸ sau pe ambele fețe (*twoside*).

Comanda care specifică tipul de document este elementul obligatoriu al *antetului* unui program L^AT_EX. Programul trebuie să aibă însă și ceea ce se numește un *corp*.

În al doilea element de meniu, i se spune, de fapt, editorului Vim cum să creeze corpul unui program L^AT_EX. Observați că, de data aceasta, este vorba despre o pereche de comenzi. Tratați-le ca pe niște paranteze. Dacă una lipsește, atunci programul conține o eroare.

Acțiunea perechii de comenzi se exercită numai asupra a ceea ce se află între ele. Lucrul acesta dă mari bătăi de cap novicilor. Creați un fișier cu extensia **tex**. Să zicem că numele acestui fișier este **prim.tex**. Deschideți **prim.tex** cu Vim. Dacă totul este în regulă, puteți rula scriptul Vim creat și aveți un meniu cu două elemente în el. Folosiți-le pentru a scrie un prim program L^AT_EX, ceva în genul a ceea ce vedeți mai jos.

```
\documentclass{article}
\begin{document}
Salut cititoarele/cititorii!
\end{document}
```

Salut cititoarele/cititorii!

program
L^AT_EX
minimal

În exemplul de mai sus vedeți în partea stângă un program minimal și-n partea dreaptă rezultatul obținut după compilare.

Ce este **compilarea**? Un program conține comenzile pe care vrem să le execute computerul. În cazul compilării, programul este prelucrat, ca să spunem așa, în întregime sa. Nu obținem rezultatul dorit pe bucăți, comandă cu comandă.

Adevărul este că L^AT_EX este suficient de flexibil pentru a lucra și

²⁷Scrieți doar pt după numărul care specifică dimensiunea literelor.

²⁸În mod tradițional, de exemplu, lucrările de licență sunt tipărite pe o singură parte a foi de hârtie. Dacă le construiți folosind tipul de document *book* va trebui să specificați explicit opțiunea *oneside*. La articole lucrurile stau exact invers.

bucată cu bucată. Scrieți în `prim.tex` doar antetul. Într-o fereastră MS-DOS, cu promptul chiar în dosarul unde este `prim.tex`, putem duce un dialog cu \TeX de genul celui care urmează:

```
E:\test\zil>latex prim.tex
This is e-TeX, Version 3.141592-2.1 (MiKTeX 2.4)
entering extended mode
(prim.tex
LaTeX2e <2001/06/01>
Babel <v3.7m> and hyphenation patterns for english loaded.
)
*\begin{document}
(D:\ed\ltx\texmf\tex\latex\base\article.cls
Document Class: article 2001/04/21 v1.4e Standard LaTeX
document class
(D:\ed\ltx\texmf\tex\latex\base\size11.clo)) (prim.aux)
*Salut lume!
*\end{document}
[1] (prim.aux)
Output written on prim.dvi (1 page, 228 bytes).
Transcript written on prim.log.
```

Promptul `>` este cel al sistemului de operare, care așteaptă o comandă în linie. Steluța este promptul sistemului \TeX , care așteaptă comenzi. Mesajele indică faptul că totul s-a terminat cu bine.

Pentru a compila fișierul `prim.tex` vă trebuie un document complet, după modelul indicat mai sus. Când dați acum comanda în linie `latex prim` nu mai trebuie să duceți nici un dialog cu \TeX . Puteți vedea rezultatul dând un dublu clic pe `prim.dvi`.

unde scriem textul eseului? Atenție, nu scrieți textul care vreți să apară pe ecran în afara corpului programului! În antet, ar fi o eroare. După `\end{}` n-ar avea nici un efect. Aceasta din urmă este eroarea tipică a novicilor. Este semnul că n-ați înțeles rolul perechii de comenzi. Nu este nici o problemă! Exersând vă dați seama care este funcția corpului programului.

Folosiți mediul integrat pentru compilare. Dați dublu clic pe fișierul `prim.tex` și acesta va fi deschis în \TeX nicCenter. Căutați pictograma **Build current file** (`CTRL+F7`). Dați un clic. În fereastra de jos a mediului integrat veți vedea o serie de mesaje. Dacă sunt `0 Error(s)`, pentru o primă încercare, totul este în ordine. Pentru a vedea rezultatul, dați un clic pe pictograma **View output**.

importanța dvi Este recomandabil să mențineți caseta **Output profile** pe opțiunea `LaTeX=>DVI` cât timp lucrați la primele versiuni ale eseului dumneavoastră. Acestea sunt fazele în care, înainte de era computerelor, ați fi scris cu mâna. Abia când pregătiți versiunea pentru

printer sau pentru ecran are sens să compilați un fișier PostScript sau pdf. Deocamdată verificați doar dacă totul este în ordine pentru toate cele trei tipuri de rezultat (dvi, ps și pdf).

Vizualizatorul fișierelor dvi este setat în așa fel la instalarea mediului integrat încât puteți da dublu clic într-un punct al textului și veți reveni în editorul de texte din mediul integrat în punctul corespunzător din sursă.²⁹ În acest fel puteți parcurge lesne ciclul compilare–vizualizare–corectare până obțineți rezultatul dorit.

Citiți deci sursa pentru a vedea dacă n-aveți erori din perspectiva limbajului L^AT_EX. Citiți rezultatul în programul de vizualizare pentru a vedea dacă el corespunde cu intențiile dumneavoastră. Corectați sursa și recompilați pentru a vedea noul rezultat.

B.1.3.1.1 Proiectul L^AT_EX Dacă elaborați un document de dimensiuni mai mari și puneți tot ce este legat de documentul respectiv într-un singur fișier, acesta devine extrem de greu de inspectat. Principiul sursei deschise ar putea funcționa și la nivelul unui fișier foarte amplu, dar caracterul deschis al sursei nu ne-ar fi de mare folos. În plus, câștigăm în planul structurării logice a documentului dacă-l sectionăm.

Ce facem însă cu sumedenia de fișiere care ar putea rezulta? Le punem într-un singur dosar. În dosarul respectiv nu punem nimic care să nu fie legat de proiectul la care lucrăm.

T_EXnicCenter este de mare ajutor când este vorba despre lucrul cu proiecte. Deschideți T_EXnicCenter și mergeți la meniul File. De aici deschideți fereastra de dialog **New Project...** Alegeți o cale (Project path) în care va sta dosarul cu proiectul dumneavoastră. Dați un nume proiectului (de pildă, „eseu“) și T_EXnicCenter va crea automat un dosar cu numele proiectului (în cazul nostru, dosarul **eseu**). În versiunea 1 Beta 6.01 a T_EXnicCenter nu este posibil decât să creați un proiect vid (empty project).

Deocamdată nu știți nimic despre B_IB_T_EX și MakeIndex, așa că puteți lăsa nebifate rubricile lor.

Dacă explorați dosarul **eseu**, veți vedea că T_EXnicCenter a creat trei fișiere: **eseu.tcp**, **eseu.tps** și **eseu.tex**. Primele două fișiere le modifică doar T_EXnicCenter. Pot fi însă lesne vizualizate cu Vim. Să vedem ce se găsește în **eseu.tcp**:

1 [FormatInfo]

2 Type=TeXnicCenterProjectInformation

²⁹Pentru deschiderea Vim din Yap v. §B.1.2.3.1.1

```
3 Version=2
4
5 [ProjectInfo]
6 MainFile=eseu.tex
7 UseBibTeX=0
8 UseMakeIndex=0
```

Pe rândurile 7 și 8 se găsesc două opțiuni evidente. Dacă la crearea proiectului n-ați bifat **Uses BibTeX** și **Uses MakeIndex**, valoarea acestor opțiuni este 0. Când veți vrea să folosiți programele respective, nu este dificil să puneți 1 în loc de 0. Nu faceți însă alte modificări.

Pe rândul 6 din fișierul **eseu.tcp** ni se spune că **eseu.tex** este fișierul principal al proiectului. Aceasta este o informație importantă. Acest fișier poate fi modificat și dinafara mediului integrat, cu ajutorul editorului Vim. Fie în mediul integrat, fie cu ajutorul Vim, puneți în fișierul principal următoarele rânduri de program **L^AT_EX**:

```
1 \documentclass[a4paper,12pt]{article}
2 \author{Prenume Nume}
3 \title{Primul meu eseu}
4 \begin{document}
5 \maketitle
6 \tableofcontents
7 \include{text}
8 \end{document}
```

Atenție în special la rândul 7 din fișierul principal. Acest rând cuprinde o comandă care cere includerea unui fișier de tip **tex**, numit **text.tex** în proiect. Nu puneți extensia fișierului în comanda **\include**. Fișierul trebuie creat de către dumneavoastră. O puteți face atât în mediul integrat, cât și-n afara acestuia. Trebuie însă neapărat să puneți acest fișier în dosarul proiectului.

Ce cuprinde fișierul **text.tex**? Deocamdată o singură linie de text pur. Este suficient atât pentru prima zi. În perspectivă, în acest fișier poate sta un întreg eseu.

Atenție! Nu puneți antet sau corp de document în fișierul acesta sau în altă parte decât în fișierul principal.

Dacă cercetați fișierul principal, vedeți că-n antet (în preambulul programului) se află o comandă care indică sistemului numele autoarei sau autorului. De asemenea, o altă comandă arată care este titlul. Aceste comenzi sunt necesare desigur pentru comanda **\maketitle** din corpul documentului.

De ce lipsesc accoladele după **\maketitle** și **\tableofcontents**? Puteți să le puneți. Nu se întâmplă nimic rău. Aceste comenzi își cu-

leg însă singure materialul. Ne vom mai întâlni, de altfel, cu comenzi cu argumentul vid.

Versiunea 1 Beta 6.20 a mediului integrat are în meniul **Project** rubrica **Create with active file as main file**. Cu ajutorul ei procesul
 5 descris mai sus se simplifică. Puteți crea direct fișierul principal și, pe baza lui, întregul proiect.

Cum compilați proiectul? Căutați pe bara cu instrumente picto-
 grama **Build Output(F7)**. Dați clic și compilați.

F7

Are rost să construim însă un proiect pentru un simplu articol,
 10 pentru un eseu de dimensiuni mici? Are foarte mult sens să facem acest lucru. Este un pas înainte în separarea procesului de elaborare a textului ca atare și a procesului de compilare. Puteți scrie fișierul **text.ttt** pe cu totul alt computer decât cel pe care compilați și doar să-l integrați ca pe un modul în proiect. De asemenea, separați
 15 comenzile privitoare structura de ansamblu a documentului de tot ce este legat de conținutul diferitelor părți ale proiectului.

În cazul unei cărți, este recomandabil ca fiecare capitol să fie pus într-un fișier aparte. Evident, în fișierul principal al proiectului, trebuie să schimbați tipul documentului din **article** în **book**.

20 S-ar putea ca unii să fie îngroziți de sumedenia de fișiere dintr-un proiect. După compilare rezultă noi fișiere. Unele dintre ele, când vă satisface rezultatul final, pot fi șterse. În T_EXnicCenter, puteți curăța proiectul de fișierele auxiliare folosind **Build** → **Clean Project**. Este recomandabil însă să aveți undeva (pe dischetă sau CD ar fi cel
 25 mai bine) tot proiectul salvat, înainte de a face curățenie.

Chiar și curățat de fișierele auxiliare, un proiect are numeroase fișiere. Cum îl distribuim pe Internet sau îl trimitem prin poșta electronică? S-ar putea ca destinatarii să aibă probleme cu descărcarea fișierelor. Soluția cea mai bună este să arhivați tot proiectul. Selec-
 30 tați dosarul proiectului. Deschideți meniul contextual. Dacă aveți 7-zip sau alt program de arhivare instalat, chemați programul de arhivare, selectați opțiunile dorite și construiți o arhivă a proiectului. Acum aveți un singur fișier, lesne de transportat.³⁰

B.1.3.1.2 Vim și sintaxa limbajului L^AT_EX Ca orice hmbaj
 35 de programare, L^AT_EX are o sintaxă care trebuie strict respectată. Dați un nume de comandă pe care L^AT_EX nu-l știe și veți primi la

³⁰Nu vă jucați cu toate posibilitățile oferite de arhivare. Există posibilitatea de a transforma arhiva într-un executabil, care se deschide printr-un simplu clic la destinație. Unele firme care asigură serviciile de poștă electronică sunt însă suspicioase și returnează toate scrisorile care au atașate programe executabile.

compilare un mesaj de eroare. Nu se poate însă face ceva pentru a depista erorile de sintaxă înainte de faza compilării.

Dacă folosiți Vim cu interfață grafică, atunci ați observat deja faptul că Vim colorează într-un mod special expresiile din limbajul \LaTeX . Comenzile sunt evidențiate prin culori. Acest lucru vă ajută să identificați erorile de sintaxă.

De asemenea, folosiți, în modul comandă normală, tasta % pentru a testa închiderea corectă a parantezelor. Dacă ați integrat și scriptul `matchit`, puteți verifica și corectitudinea perechilor de comenzi de genul `\begin{document}--\end{document}`.

Sistemul culorilor are limitele sale, dar – în practică – n-am simțit nevoia utilizării unui program special de verificare a corectitudinii sintaxei Vim. Dacă n-aveți interfață grafică, un asemenea program este însă util.

Editorul mediului integrat folosește aceeași metodă a colorării textului pentru a evidenția sintaxa programului \LaTeX . Vim mi se pare însă mai flexibil și de mai mare ajutor atunci când este vorba despre depistarea unei erori subtile de sintaxă.

B.1.3.2 Ziua a doua

Orice program \LaTeX se traduce, până la urmă, în comenzi pe care le execută procesorul \TeX . Ar fi foarte util să ne amintim că \TeX nu este un editor de texte asemenea lui Vim. Este un sistem de aranjare a textului în pagină în vederea tipăririi.

\TeX pune pe foaia de hârtie cutii (*boxes*) în care apoi pune litere.³¹

B.1.3.2.1 Modurile \LaTeX Oricine s-a jucat cu cutii știe însă că le poți fie pune unele lângă altele, fie unele peste altele. Pe foaia de hârtie, care este bidimensională, „cutiile” au doar un sens metaforic. Evident, sunt doar niște patrulatere care pot fi dispuse pe orizontală sau pe verticală.

Dacă sistemul dispune cutiile pe verticală, atunci spunem că el se află în modul vertical. Dacă sistemul așază cutii pe orizontală (pe rânduri), atunci este în modul orizontal.

Atât modul vertical, cât și modul orizontal au două variante:³²

³¹ Aici doar am reamintit un principiu de bază pentru înțelegerea modului în care funcționează procesorul \TeX . Vezi mai sus pagina 187, rândul 11.

³² Am folosit terminologia și explicațiile din Seroul[39, p.46].

vertical obișnuit În acest mod se pot pune una peste alta oricâte cutii, deoarece se trece de la o pagină la alta și așa mai departe;

vertical intern Cutiile sunt puse vertical într-o altă cutie și există o limită dată de dimensiunile cutiei în care se pun alte cutii;

- 5 **orizontal obișnuit** În acest mod se pot înșirui oricâte cutii, deoarece se trece de la un rând la altul;

orizontal strict Cutiile sunt înșiruite în altă cutie și limita este dată de cutia respectivă, dar nu se produce o trecere de la un rând la altul.

- 10 Când ne referim la „vertical“ sau „orizontal“ fără altă precizare avem în vedere variantele obișnuite ale acestor moduri.

Distincțiile din sistemul lui Knuth sunt foarte naturale, în ciuda enunțurilor abstracte de mai sus. Evident, resursele sistemului \TeX pun o limită în calea înșiruirii de *oricâte* cutii; aici era vorba însă
15 de niște distincții de principiu.

Cel de al treilea mod este cel matematic. Într-un eseu filosofic este perfect posibil să întâlnești formule logice. Acestea trebuie scrise în modul matematic. Acest mod de lucru al \TeX nu poate fi învățat însă în două zile.³³

- 20 Dacă am înțeles felul în care lucrează \TeX , atunci am înțeles și modurile de lucru ale \LaTeX . Putem, de asemenea, să facem și o distincție simplă, binară, între un mod „text“ și un mod „non-textual“. **text vs. non-text**
Textul ca atare este aranjat orizontal și vertical. Veți vedea și o serie de comenzi pentru text. Când trebuie specificat explicit că aceste
25 comenzi sunt pentru text \LaTeX , ele sunt de forma `\text{...}`.

Când pornește, \LaTeX este în modul vertical. Cum se schimbă acest mod? Prin comenzi care implică modul orizontal într-una din versiunile sale și la întâlnirea unuia dintre semnele care nu sunt rezervate.

- 30 Prin urmare, la întâlnirea unui semn care este reprodus ca atare în textul tipărit, \LaTeX trece în modul orizontal.

Pentru a vedea modurile \TeX puteți duce următorul dialog într-o fereastră MS-DOS:³⁴

```
1 >tex
2 This is TeX, Version 3.141592 (MiKTeX 2.4)
3 **\tracingcommands=1
```

³³A se vedea aici anexa B.4.

³⁴Exercițiul este pentru ucenicii vrăjitori și este sugerat de Seroul[39, p.47].

```
4 *\vbox{}
5 *a
6 *\hbox{}
7 *$x$
8 *\end
9 [1]
10 (see the transcript file for additional information)
11 Output written on texput.dvi (1 page, 260 bytes).
12 Transcript written on texput.log.
```

Chiar dacă nu știți comenzile, numele lor este sugestiv: `vbox` sugerează o cutie legată de modul vertical; iar `hbox` este o cutie care are legătură cu modul orizontal. Citirea fișierului de tip `log` este oricum instructivă:

```
1 This is TeX, Version 3.141592 (MiKTeX 2.4)
2 **\tracingcommands=1
3 *\vbox{}
4 {vertical mode: \vbox}
5 {internal vertical mode: end-group character }}
6 {vertical mode: blank space }
7 *a
8 {the letter a}
9 {horizontal mode: the letter a}
10 {blank space }
11 *\hbox{}
12 {\hbox}
13 {restricted horizontal mode: end-group character }} /
14 {horizontal mode: blank space }
15 *$x$
16 {math shift character $}
17 {math mode: the letter x}
18 {math shift character $}
19 {horizontal mode: blank space }
```

Fișierele `log` conțin mesajele sistemului pe parcursul interpretării sau compilării unui program. Este bine să ne obișnuim să le citim. Din nou, `TEXnicCenter` oferă un mare avantaj: toate mesajele sistemului apar într-o fereastră și putem naviga de la un mesaj la altul cu ajutorul unor pictograme. Puteți, de asemenea, da un clic pe programul care vă interesează și veți naviga automat către linia (în sens logic) din program care a pricinuit apariția mesajului.

B.1.3.2.2 Alineatele Dacă ați făcut tentativa de a scrie din prima zi un text cu mai multe alineate în sistemul `LATEX`, ați avut probabil o mare surpriză. Fără să știți ați intrat în modul orizontal și, dacă n-ați avut noroc, n-ați putut ieși din el. Apăsarea (o singură dată!) pe tasta `ENTER` nu vă scoate din modul orizontal.

Dacă nu ieșim din modul orizontal, nu putem construi alineate distincte. Ieșim din modul orizontal dacă apăsăm de două ori tasta ENTER. Altfel spus, un rând alb ne readuce în modul vertical.

**apăsați de
două ori
ENTER
pentru a
începe un
alineat**

Există și o comandă specială pentru crearea de alineate: `\par{}`.

- 5 N-are rost să creez aici meniuri pentru comenzi atât de simple. Este suficient să aveți un meniu pentru scheletul unei comenzi L^AT_EX.³⁵ Puneți textul la locul potrivit și ați scris comanda.

- 10 Unde punem textul alineatului? Putem să-l punem între acoladele comenzii `\par{}`? Se poate, dar n-are sens din perspectiva modurilor T_EX. Comanda `\par{}` trebuie pusă la sfârșitul alineatului. Ea spune sistemului că trebuie să treacă în modul vertical. Putem omite acoladele.

- 15 Din perspectiva clarității sursei programului L^AT_EX, recursul la rândul alb pentru a crea alineate mi se pare preferabil. Marele avantaj al faptului că apăsarea pe tasta ENTER nu este echivalentă cu trecerea la un nou alineat³⁶ este însă altul. Putem pune fiecare idee pe un rând logic distinct.

- 20 Un alineat este cărămida de bază a unui text. Fiecare alineat cuprinde un mănunchi de idei care gravitează în jurul unei idei-cheie. Este extrem de avantajos să putem pune fiecare idee pe un rând distinct.

- 25 În acest punct se vede cel mai bine de ce soluția aleasă curent în editoarele de birou este inadecvată pentru munca intelectuală. Editoarele de birou³⁷ ne forțează să scriem ideile, ca să zic așa, în modul orizontal, punându-le unele după altele. Această modalitate de scriere este avantajoasă dacă vrem să vedem imediat cum va arăta rezultatul final. Ce-ați zice însă dac-ar cere cineva ca o clădire să fie construită fără a se recurge la schele, pe motiv că acestea nu ne permit să vedem pe parcurs apropierea de rezultatul final? Dacă
30 respectăm această cerință, rezultatul final va semăna mai mult cu o colibă decât cu o clădire frumoasă.

- B.1.3.2.3 Textul invizibil** Un alineat are o idee-cheie. Putem face ceva, în sursa programului L^AT_EX, pentru a evidenția cumva ideile-cheie? Da. Editoarele de birou recurg la colorare. Această me-
35 todă este însă rezervată în sursele L^AT_EX pentru marcarea sintaxei

³⁵ A se vedea aici §A.2.2.1.1.3.

³⁶ Ideea de a nu apăsa de două ori ENTER, preluată de pildă de Funeriu[14, p.282] din Parker, este una dintre dintre absurditățile pe care le induce în mintea oamenilor utilizarea editoarelor de birou.

³⁷ Cel puțin cele pe care le știu eu.

limbajului. Există însă și o altă posibilitate: textul invizibil (în documentul tipărit în final).

comentarii În surse este bine să fie puse comentarii. Comentariile pot cuprinde, de pildă, ideea-cheie a alineatului. Comanda pentru comentarii este realizată cu ajutorului semnului rezervat `%`. Tot ce se află între simbolul procent și primul sfârșit de rând logic devine comentariu și este ignorat de către compilator. De asemenea, putem pune comentariile pe un rând separat.

Dacă examinăm începuturile de rând din sursa alineneatului de la pagina 205, rândul 22 vedem un comentariu (pe rândul al doilea în extrasul de mai jos) care nu apare în text după compilare (rolul lui este de a-mi permite să sesizez rapid esențialul alineatului):

```
1 În acest punct se vede cel mai bine de ce soluția aleasă (...)
2 %editoarele de birou fac obscură logica alineatului
3 Editoarele de birou ne forțează să scriem ideile (...)
4 Această modalitate de scriere este avantajoasă dacă (...)
5 Ce-ați zice însă dac-ar cere cineva ca (...)
6 Dacă respectăm această cerință, rezultatul final (...)
```

Până la urmă, din punctul de vedere al textului invizibil, diferența între editoarele de birou și `LaTeX` este una mai mult de abordare decât de substanță. Editoarele de birou au posibilitatea de a insera text invizibil la tipărire. Vim sau alte editoare adaptate cerințelor `LaTeX` colorează într-un mod aparte comentariile.

B.1.3.2.4 Secțiunile Este bine să folosim cât mai mult comentariile. În cele ce urmează vom explica felul în care pot fi create secțiunile unui eseu într-un document `LaTeX`. Titlul lor nu ne spune însă, după o vreme, chiar totul despre ce am vrut să punem în secțiunea respectivă. Un comentariu ne ajută enorm din acest punct de vedere.

În `LaTeX`, articolele sunt divizate în secțiuni, subsecțiuni, subsubsecțiuni, paragrafe și subparagrafe. Nu există capitole în cazul unui articol.

Care sunt comenzile pentru a crea aceste cinci tipuri de secțiuni? Comenzile pot fi desprinse din următoarele linii de cod pentru meniuri Vim:

```
1 imenu ltx2.section \subsection{<CR>\label{<Up><Right>
2 imenu ltx2.subsection \subsubsection{<CR>\label{<Up><End><Left>
3 imenu ltx2.subsubsection \paragraph{<CR>\label{<Up>
4 <End><Left>
5 imenu ltx2.paragraph \subparagraph{<CR>\label{<Up><End><Left>
```

```
6 imenu ltx2.subparagraph \subparagraph{<CR>\label{<Up><End><Left>
```

Puneți titlul secțiunii între acoladele primei comenzi din perechea de comenzi. Sensul celei de a doua comenzi îl vom explica mai jos.

Termenii din limba engleză pentru ultimele două tipuri de secțiuni sunt susceptibili să genereze confuzii. În limba engleză, un *paragraph* este mai degrabă un alineat. Numele comenzilor sunt mai puțin importante însă. Semnificativ este că acestea sunt toate secțiuni ale unui text. Cea mai mică secțiune grupează un număr de alineate, centrate în jurul unei idei-cheie.

Comenzile de secționare pot fi date și-n mediul integrat TeXnic-
 10 Center. Recomandarea noastră este să NU faceți acest lucru. Mediul integrat construiește automat argumentul comenzii `\label{}` și pune o secvență `sec`: la începutul oricărui argument. Acest obicei mai mult încurcă însă lucrurile. Cel mai bine ar fi ca argumentul din comanda `\label{}` să fie un unic cuvânt. Dacă puneți două puncte
 15 în argument, atunci nu se mai respectă această cerință.

La ce slujește comanda `\label{}`? Scrieți comenzi de secționare în fișierul `text.tex` al proiectul `eseu`. Nu le scrieți în fișierul principal. Scrieți comenzi pentru toate cele cinci tipuri de secțiuni, cu titluri și ceva text în fiecare secțiune. Transformați însă în comentarii rândurile unde se află comenzile `\label{}`. După ce compilați, studiați efectul obținut. În textul generat astfel sunt vizibile secțiunile. Primele trei tipuri au și numere puse automat. Eliminarea comenzilor `\label{}` n-a stricat însă nimic.

În limba engleză, *label* înseamnă „etichetă”. Secționarea textului
 25 nu depinde de etichete. Etichetele au cu totul alt rol decât acela de a secționa textul. De fapt nu au, în sine, nici o legătură cu secțiunile ca atare. Ele sunt marcaje, sunt punctele pe care le caută alte comenzi din program. Dacă vrem să trimitem la o secțiune din text, avem nevoie de aceste marcaje speciale.

30 Cum ar fi bine să arate aceste etichete? Voi folosi un exemplu chiar din textul de față:

```
\subsubsection{Învățarea Vim într-o zi}
\label{InvatareaVimIntroZi}
```

Cred că acum sunteți convinse și convingi că este o doză de ironie în spatele titlului luat drept exemplu. Vim nu se învață într-o zi și, cu atât mai puțin, ~~La~~TeX. Lucrul acesta este însă aici mai puțin
 35 semnificativ. Ce se vede însă foarte bine este felul în care în etichetă au fost eliminate spațiile, diacriticele și, de fapt, orice alt element care ar putea genera confuzii la căutarea prin text.

**creați
secțiunile
cu Vim**

label

De ce pledez atât de mult pentru etichetele simple, compacte? Practic, gândiți-vă c-ați construi un șablon, o expresie regulată pentru a căuta etichete într-un program \LaTeX . Nu este prea greu să izolezi conținutul aflat între acoladele comenzii `\label{}`. Este însă mult mai simplu să lucrezi după aceea cu el dacă este un simplu cuvânt. 5

\TeX nicCenter dispune de o fereastră pentru vizualizarea structurii proiectului. De altfel, face acest lucru foarte bine, cu sau fără etichete. Mediul integrat nu se bazează pe ele pentru a explora structura proiectului. 10

În \TeX nicCenter, numai dacă avem un proiect putem explora structura documentului, indiferent de numărul de fișiere. Este un alt argument în favoarea creării unui proiect chiar și-n cazul unui simplu articol.

În cazul unei cărți există două tipuri suplimentare de secțiuni. 15
Le vom prezenta tot sub forma codului pentru meniuri Vim:

```
1 imenu ltx2.part \part{<CR>\label{<Up><Right>
2 imenu ltx2.chapter \section{<CR>\label{<Up><Right>
```

Pentru a nu primi mesaje de eroare, trebuie să modificați în antetul fișierului principal al proiectului tipul de document. Acesta trebuie să fie `book`.

O carte poate avea părți și capitole. Acesta este sensul acestor 20
tipuri de secțiuni ale unui document.

§

Aici preferăm să ne referim la toate aceste diviziuni ale unui document folosind un număr precedat de simbolul §. Pentru a produce acest semn este nevoie de o comandă specială: `\textsection`.

B.1.3.3 Ziua a treia 25

Atunci când lucrăm cu un editor de birou executăm o serie întreagă de operații uzuale de prelucrare a unui text. Acestor operații propun să le consacrați ziua a treia de învățare a \LaTeX .

Pentru început studiați câteva meniuri simple:

```
1 imenu ltx3.emph \emph{<Left>
2 imenu ltx3.textit \textit{<Left>
3 imenu ltx3.underline \underline{<Left>
4 imenu ltx3.textbf \textbf{<Left>
5 imenu ltx3.texttt \texttt{<Left>
6 imenu ltx3.textsc \textsc{<Left>
7 imenu ltx3.textsf \textsf{<Left>
8 imenu ltx3.textsl \textsl{<Left>
```


Prima dintre comenzile de mai sus este probabil cea mai importantă. În ciuda aparențelor, L^AT_EX face mult mai multe lucruri în mod automat decât un procesor de cuvinte. Prima comandă îi spune sistemului L^AT_EX să evidențieze porțiunea de text cuprinsă în acoladele comenzii (argumentul comenzii), în funcție de context. L^AT_EX alege metoda potrivită, nu utilizatoarea. Este metoda recomandabilă de evidențiere a textului. În anumite puncte ale textului, alte metode nu funcționează sau nu funcționează cum vă așteptați.

Pentru a vedea cum funcționează comenzile de mai sus cel mai potrivit este să dăm un exemplu:

```
\emph{Eseul filosofic}\\
\textit{Eseul \emph{filosofic}}\\
\underline{Eseul \emph{filosofic}}\\
\textbf{Eseul \emph{filosofic}}\\
\texttt{Eseul \emph{filosofic}}\\
\textsc{Eseul \emph{filosofic}}\\
\textsf{Eseul \emph{filosofic}}\\
\textsl{Eseul \emph{filosofic}}\\
\textrm{Eseul \emph{filosofic}}
```

```
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
```

stilurile
literelor

Vă întrebați acum, de bună seamă, ce rol au cele două bare oblice inverse. Ele îi spun sistemului că trebuie să forțeze trecerea pe alt rând.

Ce facem dacă vrem, din contră, să evităm ruperea rândului. Dacă este vorba despre un spațiu alb, atunci punem în locul spațiului alb o tildă, semnul ~.

Este momentul acum să arătăm și cum se obțin semnele rezervate. La urma urmei, ele sunt semne uzuale pentru un editor de birou. Dacă studiați exemplul următor, vedeți că nu sunt greu de obținut nici în L^AT_EX:

```
100\${}\n
XYZ\&{}\co.\n
\#\n
100\%\n
meniu\_vim
```

```
100$
XYZ&co.
#
100%
meniu_vim
```

semne
rezervate

Tehnica folosită este cea a transformării simbolurilor rezervate în nume de comenzi L^AT_EX. Chiar dacă argumentele acestor comenzi rămân vide, are sens să păstrăm acoladele. În acest fel comunicăm sistemului L^AT_EX unde se termină numele comenzii.

Terminarea numelui unei comenzi poate fi indicată și printr-un spațiu alb. Sursa programului L^AT_EX devine însă mai lizibilă prin folosirea acoladelor.

În cele ce urmează vom vedea și o excepție de la regula transformării semnelor rezervate în nume de comenzi.

semne
rezervate
(continuare)

```
\{{}de aici\\
până aici\}{}\\
\^{}Z\\
\texttt{\^{}{}}\\
\textbackslash{}label\{\\
```

```
{de aici
până aici}
^Z
~
\label{}
```

Bara oblică inversă este obținută, în modul text, cu comanda `\textbackslash{}`.

5

\LaTeX este sensibil la diferența de stil între semnele folosite în modul matematic și-n modul orizontal, obișnuit sau strict. Acest lucru se vede din prezența unor comenzi specifice textului. Unele le-am menționat deja. Altele pot fi studiate în exemplele care urmează.

semne
speciale

```
\textgreater{}\\
\textless{}\\
\textsection{}\\
\textparagraph{}\\
\textcopyright{}\\
```

```
>
<
§
¶
©
```

10

Pentru a ușura comparațiile am inclus în cazul ghilimelelor și versiunile scurte ca să zicem așa.³⁸

ghilimele

```
\textquotedblleft{}\\
“\\
\textquotedblright{}\\
”\\
\textquoteleft{}\\
‘\\
\textquoteright{}\\
’
```

```
“
“
”
”
‘
‘
’
’
```

Mai există și alte comenzi importante în \LaTeX pe care s-ar putea să le folosim relativ frecvent:

```
\ldots{}\\
-\\
--\\
---
```

```
...
-
--
---
```

15

Din exemplul de mai sus rezultă implicit cum putem spune sistemului să nu rupă punctele de suspensie. Ce facem însă în cazul ortogramelor românești? La trecerea de pe un rând pe altul s-ar putea să ne ciocnim de ruperea ortogramei într-un punct nedorit. Felul în care putem proceda ne este sugerat de exemplele următoare:

20

³⁸ Atenție! Acestea sunt ghilimele standard în limba engleză. Ghilimelele românești trebuie construite, după cum vom arăta mai jos.

```

intr-o intr-o intr-o intr-o intr-o
intr-o parte intr-o zi zi
intr-o intr-o
intr-o
intr-o

```

```

intr-o intr-o intr-o intr-o \hbox{intr-o}
intr-o parte intr-o zi zi
\hbox{intr-o} \hbox{intr-o}
\hbox{intr-o}
\hbox{intr-o}

```

```

intr-o intr-o intr-o intr-o intr-
o intr-o parte intr-o zi zi intr-o
intr-o intr-o intr-o

```

ortograme

```

intr-o intr-o intr-o intr-o intr-o
intr-o parte intr-o zi zi intr-o
intr-o intr-o intr-o

```

Există o subtilitate în utilizarea `\hbox{}`. Conținutul cutiei este în mod orizontal strict, dar comanda nu face trecerea la modul orizontal obișnuit în locul în care este plasată. Putem compara alte două exemple:

```

\hbox{intr-o}
\hbox{intr-o}
\hbox{intr-o}

\mbox{intr-o}
\mbox{intr-o}
\mbox{intr-o}

```

```

intr-o
intr-o
intr-o

```

```

intr-o intr-o intr-o

```

În manualul său[21], creatorul L^AT_EX, Leslie Lamport, recomandă folosirea comenzii `\mbox{}` pentru evitarea rupeerii textului.³⁹

B.1.3.3.1 Blocurile Există, după cum am văzut, comenzi fără acolade puse după ele. Sunt comenzi care fie nu au nevoie de nici un material (argumentul lor este efectiv vid), fie își iau singure materialul pe care-l prelucrează.

O comandă precum `\today` afișează data curentă. Putem să ne dispensăm de acoladele puse după această comandă.

Este oare posibil să folosim, în limbajul L^AT_EX, acolade care nu sunt precedate de bara oblică inversă și numele comenzii? Răspunsul este afirmativ. Acoladele separă un *bloc* din programul L^AT_EX.

Care este rostul blocurilor? Ideea de bază este cea a localizării anumitor proprietăți ale textului. Cea mai simplă ilustrare este legată de evidențierea unei porțiuni de text. Să examinăm un exemplu foarte simplu:

```
Eseul{\em filosofic}
```

```
Eseulfilosofic
```

Rolul acoladelor este să delimiteze o porțiune de *text*. Imediat după prima acoladă urmează o declarație. Declarația este de fapt tot o comandă. Ați putea pune chiar și acolade după ea, dar cred că aici acest stil de a scrie programul nu-și are rostul. Spațiul alb este preferabil.

³⁹ A se vedea chiar listele volante cu comenzi uzuale, pe care le găsim în carte.

Declarațiile au efect numai în limitele unui bloc de program \LaTeX .⁴⁰

Este extrem de ușor să construim un meniu Vim pentru a insera blocuri și declarații. Dăm aici doar rândul de cod pentru evidențierea textului:

```
1 imenu ltx3.em {\em }<Left>
```

A se observa poziția la care se întoarce cursorul Vim. Rămâne un spațiu alb între declarație și ce punem în blocul respectiv.

Ce am mai putea declara? Putem specifica forma (stilul) tipurilor de litere.⁴¹ Dacă faceți exerciții cu exemplul care urmează, veți descoperi lesne rostul următoarei serii de declarații:

declarații

```
{\itshape Eseul filosofic}\\
{\scshape Eseul filosofic}\\
{\slshape Eseul filosofic}\\
{\upshape Eseul filosofic}\\
{\bfseries Eseul filosofic}\\
{\mdseries Eseul filosofic}\\
{\ttfamily Eseul filosofic}\\
{\sffamily Eseul filosofic}\\
{\rmfamily Eseul filosofic}
```

<i>Eseul filosofic</i>
ESEUL FILOSOFIC
<i>Eseul filosofic</i>
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic
Eseul filosofic

Se observă că există o corespondență între comenzile prin care se dă o anumită formă literelor din text⁴² și declarații.⁴³

B.1.3.3.2 Revizia textului programelor \LaTeX Se întâmplă de multe ori să lucrez și să uit să subliniez o literă sau să scriu cursiv un cuvânt. Când fac revizia textului descopăr aceste probleme. Ce-i de făcut? Mi-am construit meniurile care urmează tocmai pentru a rezolva genul acesta de probleme.

```
1 imenu ltx3.cuvantInAcolade {\<Esc>ea}
2 imenu ltx3.cuvantInComanda {\<Esc>ea}
3 imenu ltx3.cuvantTextsc \textsc{\<Esc>ea}
4 imenu ltx3.cuvantTexttt \texttt{\<Esc>ea}
5 imenu ltx3.cuvantTextit \textit{\<Esc>ea}
6 imenu ltx3.cuvantSubliniat \underline{\<Esc>ea}
7 imenu ltx3.litSubliniata \underline{\<Esc>la}
8 imenu ltx3.litInAcolade {\<Esc>la}
```

⁴⁰Termenul „declarație” este cel folosit chiar de Leslie Lamport. A se vedea Lamport[21, p.27].

⁴¹Același *font* poate fi normal sau cursiv. aldin și așa mai departe.

⁴²Vezi mai sus pagina 209, rândul 11.

⁴³Observația aceasta am preluat-o de la Lamport[21, p.37].

Felul în care funcționează meniurile este destul de evident. Treceți Vim în modul `insert`. Puneți cursorul în fața cuvântului sau a literei. Dați comanda și asta este tot. Restul este doar exercițiu.

Comenzile de mai sus au însă un rol destul de limitat. S-ar putea să am cu Vim divergențe în privința ideii de cuvânt. Să modific ceea ce înțelege Vim prin cuvânt? Uneori ar fi absurd să fac așa ceva. Pentru mine, `text.tex` reprezintă un singur cuvânt. Din păcate, eu folosesc înțelesul termenului respectiv. Vim se orientează după punctul pus în numele fișierului. N-am însă nici un interes să schimb felul în care Vim înțelege rolul punctului.

Toată lumea știe că editoarele de birou au o soluție simplă la problema de mai sus: selectezi textul și apeși pe un buton virtual; porțiunea selectată își schimbă forma pe ecran. Nu este chiar așa de greu să construim meniuri Vim care ne permit să lucrăm cu porțiuni de text selectate.

```
1 menu ltx3n.sellnAcolade xi{<Esc>pa}
2 menu ltx3n.selInComanda xi\{<Esc>pa}
3 menu ltx3n.selTextsc xi\textsc{<Esc>pa}
4 menu ltx3n.selTexttt xi\texttt{<Esc>pa}
5 menu ltx3n.selTextit xi\textit{<Esc>pa}
```

Aceste meniuri trebuie studiate ceva mai atent decât cele de mai sus. Ele nu sunt active în modul `insert` și este firesc să fie așa. Atunci când selectați text, Vim își schimbă modul. Când exersați, acordați atenție și deosebirilor dintre efectul comenzilor în modul `(insert)SELECT` și modul vizual, precum și rolului punctului din care începeți selecția.

Cred că acum vestea proastă și vestea bună în legătură cu L^AT_EX (și Vim!) au prins deja contur. Vestea proastă este c-ar trebui să dobândiți un minimum de deprinderi de programatoare sau programator. Vestea bună este legată de flexibilitatea sistemului și de posibilitatea de a face (în principiu) orice cu textul tipărit.

Felul în care sunt percepute cele două vești depinde în cea mai mare parte de câtă programare știți deja. Dacă știți programare, s-ar putea să fiți uimită sau uimit să vedeți câte lucruri se pot face fără structuri condiționale, funcții, obiecte și altele. Dacă nu știți, s-ar putea să nu apreciați corect cât de multe lucruri face sistemul pentru dumneavoastră și cât de puțin investiți, de fapt, în raport cu ceea ce primiți.

B.1.3.4 Ziua a patra

Prin ce ai deosebi dintr-o privire un eseu academic (filosofic sau nu) de o lucrare de factură neacademică (un articol de ziar, de exemplu)? Notele de subsol, uneori foarte abundente, disting imediat eseu academic de lucrările neacademice.

notele de subsol Producerea impecabilă de note de subsol este unul dintre atuurile \LaTeX . Comanda prin care sunt realizate notele de subsol este foarte simplă. Iată codul pentru un meniu Vim în modul insert:

```
1 imenu ltx4.Nota\ de\ subsol \footnote{<Left>
2 imenu ltx4.Nota\ marginala \marginpar{<Left>
```

După cum se vede din a doua linie de cod de mai sus, comanda pentru nota care apare pe margine este asemănătoare cu aceea pentru nota de subsol. Dacă este vorba despre un document tipărit pe ambele fețe ale foii (o carte), atunci \LaTeX plasează automat nota pe marginea exterioară. O subtilitate a notei marginale o reprezintă posibilitatea de a folosi două valori pentru argumentul comenzii: una pentru marginea din stânga, alta pentru cea din dreapta. Puteți studia diferența plasând comanda `\marginpar[stânga]{dreapta}` în diverse puncte ale textului, în așa fel încât s-apară când pe pagini pare, când pe pagini impare. Nu lăsați una dintre paranteze fără conținut; sistemul va interpreta paranteza goală ca pe o comandă și nu va afișa nimic.

B.1.3.4.1 Acolo unde scriau tipografii Închipuiți-vă că sunteți într-o tipografie veche. Bătrânul meșter tipograf culege textul literă cu literă. Caută prin cutiile din fața sa litere și formează rânduri. La capăt de rând desparte-n silabe, dacă este cazul, și trece pe rândul următor. Și tot așa până termină un alineat și apoi întreaga pagină de carte. Apare însă o problemă. Textul rândurilor nu se termină exact în marginea din dreapta. Arată mai mult ca textul dactilografiat decât ca un text tipărit.

Într-o altă formă, problema apărea și-n cazul articolelor de ziar. Articolului i se rezerva un anumit spațiu în pagina de ziar. Se întâmpla însă ca articolul să fie un pic prea scurt și să nu umple bine locul rezervat. Nu-i nimic, spuneau tipografii. *Aici scriem noi!*

Tipografii „scriau“, cu spații albe. Se pune „albitură“. În zilele noastre un program de așezare în pagină a textului, cum este \LaTeX , scrie și el precum vechii tipografi.

În mod normal, \LaTeX produce o aliniere a textului atât la marginea din stânga, cât și la cea din dreapta. Uneori nu reușește să

încadreze exact textul între margini sau în spațiul vertical alocat și compilatorul se plânge de existența unor bad boxes. Acesta este cel de al treilea tip de mesaje pe care le primim de la compilator. Putem primi mesaje cu privire la erori (de sintaxă L^AT_EX),
 5 avertismente (de exemplu, ni se atrage atenția că trebuie să recompilăm sursa) sau aceste mesaje cu privire la textul care nu încapă bine în cutii.

Dacă are prea mult text pe un rând, L^AT_EX spune că a apărut o `overfull \hbox..` Dacă are prea puțin text pe un rând, L^AT_EX
 10 spune că a apărut o `underfull \hbox..` Să studiem puțin exemplul care urmează.

Cuvinte magice pentru ucenicii
 vrăjitori când
 nu reușesc să compileze
 programul:ab:rasintaxadabra
 ab:racompiladabra

Cuvinte magice pentru ucenicii vrăjitori când nu reușesc să compileze pro- gramul:ab:rasintaxadabra ab:racompiladabra

*bad
box(es)*

Inițial sistemul s-a plâns c-au apărut cutii orizontale prea puțin umplute. El nu știe unde vreau să folosesc două puncte pentru a
 15 introduce o enumerare și unde cele două puncte fac parte din expresia magică. Am pus un spațiu alb după două puncte, acolo unde introduc enumerarea. Rezultatul nu a fost total reușit. Sistemul nu știa cum să despartă-n silabe cuvinte magice. Comanda `\-` îi spune unde **poate** separa silabele. Acum situația s-ar îndrepta automat
 20 dacă sistemul ar ști că textul este în română. În prima săptămână, este mai bine să lăsați sistemul setat doar pentru limba engleză. Dacă faceți exerciții cu un exemplu similar, forțați despărțirile în silabe după regulile limbii române. Aveți mai jos exemplul cuvântului „uceni\ -cii”.

Cuvinte magice pentru
 uceni\ -cii
 vrăjitori când
 nu reușesc să compileze
 programul: ab:rasin\ -taxadabra
 25 ab:racompiladabra

Cuvinte magice pentru uceni- cii vrăjitori când nu reușesc să compileze programul: ab:rasin- taxadabra ab:racompiladabra

Cele două casete arată total diferit. Prima este dezastruoasă. Cuvintele sunt dispuse cu vădită stângăcie. A doua este mult mai echilibrată.

Putem scrie, precum tipografii, cu spații albe pe orizontală sau pe
 30 verticală? Desigur că da. La urma urmei, în L^AT_EX, putem programa orice acțiune pe care ar executa-o un tipograf de modă veche.

Comenzile folosite sunt `\hspace{}` și `\vspace{}`. Între acolade

trebuie indicată o valoare numerică, fie în unități de măsură uzuale, fie în unități de măsură tipografice.

Dacă studiați puțin vizualizatorul Yap al mediului integrat, vedeți că este capabil să afișeze atât date cu privire la sursă, cât și cu privire la paginile pregătite pentru tipar. Mișcați cursorul și, de asemenea, schimbați pagina la care vă aflați pentru a sesiza felul în care se schimbă aceste date. Poziția cursorului pe pagină este indicată pe coordonatele carteziene. Unitățile de măsură folosite sunt numite „puncte”. Yap folosește prescurtarea pt.

Acum putem ilustra pe un exemplu introducerea de spațiu alb pe orizontală sau pe verticală. Practic, scriu un text, compilez, vizualizez, măsoar cu ajutorul Yap și introduc comenzile pentru a scrie „precum tipografii”.

spații albe

```
Eseul filosofic Eseul filosofic\\
Eseul\hspace{68pt} filosofic\par
\vspace{10pt}
Eseul filosofic
```

Eseul filosofic	Eseul filosofic
Eseul	filosofic
Eseul filosofic	

Modul de lucru descris mai sus NU este însă recomandabil. El folosește partea cea mai discutabilă din WYSIWYG: ideea de a ne plimba cu cursorul pe ecran și de a aprecia „din ochi” poziția în care plasăm textul.

L^AT_EX oferă posibilitatea unor calcule precise. Ar trebui să ne facem un proiect de aranjare în pagină. S-ar putea însă ca ziua a patra să aibă o valoare simbolică: suntem încă între lucrul aproximativ (și comod) și acribia L^AT_EX.

B.1.3.4.2 Mediile L^AT_EX Ce facem însă dacă vrem ca textul să fie aliniat doar la marginea din stânga? Sau dacă dorim să-l aliniem doar la dreapta? Sau să-l centram. Trebuie să-i spunem tipografului cum să aranjeze textul.

Pentru a rezolva problema de mai sus putem folosi *mediile* L^AT_EX. Mediile nu sunt altceva decât o pereche de comenzi `\begin{}` `\end{}`. Perechea aceasta de comenzi se comportă exact ca o pereche de paranteze. Numele mediului este trecut între acolade și trebuie să fie același după `begin` și după `end`.

La ce slujește un mediu? Să zicem că vrem să avem un alineat sau mai multe cu text centrat. Atunci împarantezăm textul pe care vrem să-l centram între comenzile unui mediu.

Primul exemplu de medii va fi cel pentru centrarea textului, respectiv alinierea la stânga sau la dreapta.


```
\begin{center}Eseul\end{center}
```

```
\begin{flushleft}
```

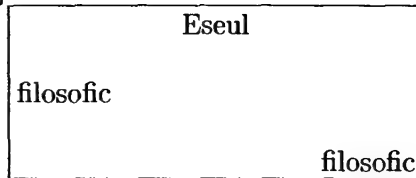
```
filosofic
```

```
\end{flushleft}
```

```
\begin{flushright}
```

```
filosofic
```

```
\end{flushright}
```



5 Dacă studiați atent exemplele de mai sus, vedeți că toate cele trei medii, când se încheie efectul lor, fac trecerea la modul vertical. Nu trebuie însă să trageți de aici concluzia că orice mediu face trecerea la un nou alineat atunci când se termină.

Cum am putea construi meniuri Vim pentru a scrie mai ușor comenzile pentru medii? Mai jos sunt oferite două genuri de soluții.

```
1 imenu ltx4.mediuOriz \begin{}\end{}\<Esc>3ba
2 imenu ltx4.mediuVer \begin{}\<CR>\end{}\<Esc>2<Up><Right>a
3 imenu ltx4.mediu <Esc>:let nume=inputdialog("Mediul?\n itemize
4 \n enumerate\n description\n flushleft\n center\n quote
5 \n quotation\n verse\n verbatim ")<CR>:let mediu="\begin{" .nume."}
6 \n\n\end{" .nume."}"<CR>:put=mediu<CR><Down><Home>i
```

10 Prima soluție are două versiuni. Mai întâi este creat un meniu care ne ajută să inserăm un mediu într-o linie de program. A doua versiune a primului gen de meniu ne permite să scriem însă într-un mod mult mai lizibil codul aferent unui mediu.

Problema primului gen de meniuri este aceea că trebuie să scriem de două ori numele mediului. Aceasta este adesea o sursă de erori. Uneori uiți să scrii numele mediului în acoladele lui end. Alteori
15 greșești ceva într-una dintre instanțele numelui.

Al doilea gen de meniuri încearcă să rezolve problema de mai sus. Prețul plătit este un cod Vim ceva mai complicat. Am redat doar varianta de scriere pe verticală a mediului în a doua versiune. Acest meniu face să apară o casetă de dialog. Există chiar și o listă
20 cu nume de medii. Ea are doar un caracter orientativ. Puteți să scrieți ce nume vreți în caseta de dialog. Trebuie dat apoi un clic pe butonul OK al casetei și mediul va apărea în fișier. Nu rămâne decât să puneți ceva între începutul și sfârșitul mediului.

Nu uitați că rândurile 3-6 reprezintă un singur rând logic Vim.
25 Mărimea paginii de hârtie, ca și ecranul, ne obligă să fragmentăm acest rând în mai multe rânduri vizuale.⁴⁴

Cu ajutorul mediilor pot fi create diverse tipuri de liste. Sintaxa comenzilor este lesne de-nțeles dacă studiați exemplul care urmează.

⁴⁴Cred că este mai comod să folosiți situl cărții, de unde puteți prelua codul ca atare, fără să fiți nevoiți să retastați totul.

```
\begin{itemize}
\item Primul pe listă
\item Al doilea pe listă
\item Al treilea pe listă
\end{itemize}
\begin{enumerate}
\item Primul pe listă
\item Al doilea pe listă
\item Al treilea pe listă
\end{enumerate}
\begin{description}
\item [Primul] pe listă
\item [Al doilea] pe listă
\item [Al treilea] pe listă
\end{description}
```

- Primul pe listă
 - Al doilea pe listă
 - Al treilea pe listă
1. Primul pe listă
 2. Al doilea pe listă
 3. Al treilea pe listă

Primul pe listă

Al doilea pe listă

Al treilea pe listă

Alte medii ne permit introducerea de citate în text sau a versurilor. Pentru citate scurte este potrivit mediul `quote`. Pentru citatele mai lungi mediul `quotation` este mai adecvat; el formează citatul ca pe un text de sine stătător.

\LaTeX , care este standardul *de facto* când este vorba despre tipărirea de cărți de informatică, are desigur și posibilități de reproducere a programelor sau fragmentelor de program. Liniile de program pe care le vedeți în această carte sunt realizate cu mediul `verbatim`. Mediul acesta poate fi folosit însă ori de câte ori vrem să dăm unei porțiuni de text aerul de pagină dactilografată.

B.1.3.4.3 Modul matematic \LaTeX este vestit mai ales pentru capacitățile sale de a produce formule matematice. Oricât de repede am trece în revistă sintaxa limbajului \LaTeX , nu se poate să nu spunem câteva cuvinte despre modul matematic.

Chiar și-n texte care nu sunt de factură matematică, vrem adesea să folosim indici. Pentru a face acest lucru în \LaTeX trebuie să trecem sistemul în modul matematic. Dacă folosim mediul `math` putem insera fragmentul de text matematic într-un alineat uzual. Mediul `math` face mai întâi trecerea de la modul vertical sau orizontal la modul matematic și, la ieșire, revine în modul orizontal.

Să vedem practic cum putem scrie indici:

```
Indice suprascris
\begin{math}x^i\end{math}\\
Indice subscris
\begin{math}x_j\end{math}\\
Doi indici
```

Indice suprascris x^i

Indice subscris x_j

Doi indici x_{ij}

Arsenalul de simboluri care pot fi folosite în modul matematic este impresionant. Săgețile, de pildă, la care am recurs și noi pe parcurs sunt inserate cu ajutorul modului matematic.

`A\rightarrow{}B\hspace{50pt}`

`A\Rightarrow{}B\hspace{50pt}`

`B\leftarrow{}A\hspace{50pt}`

`B\Leftarrow{}A`

$A \rightarrow B$	$A \Rightarrow B$
$B \leftarrow A$	$B \Leftarrow A$

- 5 Există și un mod prescurtat de a nota modul matematic care servește la inserarea de formule în rândurile obișnuite de text. Este important să menționăm această notatie prescurtată pentru că ea dezleagă misterul încă unuia dintre cele zece simboluri speciale din L^AT_EX. În loc de `\begin{math}\end{math}` putem scrie doar `$$`.

- 10 Modul matematic ne va dezvălui secretele încă unuia dintre semnele rezervate. Semnul `&` servește la realizarea de tabele. Iar fără tabele nu putem, de fapt, scrie un text matematic ceva mai complicat. O matrice, de exemplu, trebuie construită ca un tabel. Exemplul unei matrici foarte simple încheie, de altfel, incursiunea aceasta în
- 15 zona scrierii matematice.

`\begin{displaymath}`

`\begin{array}{ccc}`

`1 & 0 & 1 \\\`

`0 & 1 & 0 \\\`

`1 & 0 & 1`

`\end{array}`

`\end{displaymath}`

1	0	1
0	1	0
1	0	1

matrice

- Exemplul necesită câteva comentarii. N-am primi de la compilator un mesaj de eroare dac-am folosi mediul `math`. Pentru afișarea de sine stătătoare a unei formule trebuie să folosim însă mediul
- 20 `displaymath`. Textul matematic nu mai este încastrat în alineat, lucru care nu prea ar avea, de altfel, sens în cazul matricii.

Mediul `array` nu poate fi invocat decât în modul matematic. Cele sau cei care au învățat limbajul Pascal la informatică știu desigur că `array` are sensul tehnic de tabel.

- 25 În cazul mediului `array` trebuie să indicăm nu doar numele mediului, ci și formatarea coloanelor. De aici prezența imediat atunci când începe mediul a unei perechi de acolade. Litera `c` indică faptul că textul de pe coloană trebuie centrat. Sunt trei litere pentru că sunt trei coloane.

- 30 Separatorul coloanelor este semnul `&`. Folosirea celor două bare oblice inverse este aidoma celei din modul text. Ele comandă trecerea la un nou rând.

B.1.3.5 Ziua a cincea

Ziua anterioară a fost încărcată, așa c-am rezervat pentru astăzi elementul care încoronează posibilitatea de a scrie eseuri în stil academic: sistemul trimitărilor. Trimiterile fac posibilă lectura neliniară a eseurilor mai ample.

5

B.1.3.5.1 Trimiterile în limbajul \LaTeX Comanda `\label{}` am amintit-o deja, în contextul discuției despre comenzile de secționare. Rolul ei nu a fost însă discutat.

În limba engleză, `label` înseamnă *etichetă*. Comanda `\label{}` nu este singurul mod de a construi etichete în lumea mai largă a sistemului \LaTeX . Vă veți întâlni cu etichete și atunci când folosiți baze de date de tip `B \LaTeX` . Rolul unei etichete este de a marca un loc în text. Locul respectiv se află într-o anumită secțiune a textului, pe o anumită pagină sau pe un anumit rând. Ne putem folosi de el pentru a trimite la secțiunea, la pagina sau la rândul respectiv.

10

15

De ce nu folosim numărul secțiunii? Motivul este foarte simplu. Numărul secțiunii se poate schimba, dacă modificăm structura textului. Folosim etichete unice. Chiar dacă locul lor în text se schimbă, sistemul poate stabili unde se află în momentul compilării. Atenție doar la faptul că, în cazul etichetelor, este nevoie de cel puțin două compilări. La o primă compilare sistemul stabilește locul etichetelor în text. La a doua compilare sistemul face legăturile între trimiteri și locul către care se face trimiterea.

20

`\ref{}`

Care sunt comenzile pentru trimiteri? Pentru a trimite la o secțiune din text se folosește comanda `\ref{}`. Argumentul comenzii este o etichetă. Comanda returnează numărul secțiunii (unde este plasată eticheta). În fața numărului trebuie să puneți expresia dorită. Puteți folosi, de pildă, pe `\textsection{}`, care va produce simbolul §.

25

`\pageref{}`

Pentru a ne referi la o pagină trebuie să folosim o altă comandă, `\pageref{}`. Argumentul comenzii este tot o etichetă. Spre deosebire de `\ref{}`, rezultatul lui `\pageref{}` este un număr de pagină.

30

Sistemul descris mai sus are un caracter general. Pe măsură ce învățați mai multe despre bibliografie, figuri, tabele, veți vedea că același mecanism este folosit pentru a ne referi la intrări din bibliografie, figuri sau tabele.

35

Cu tot zgomotul în jurul hipertextului și a paginilor de web, acestea nu fac decât să exploateze pe larg ideea trimitărilor. Ceea ce era folosit doar în lucrările academice a ajuns să fie utilizat de

către toată lumea. Senzația produsă de aceste trimiteri pe Internet este una mai ciudată, pentru că, în vreme ce o trimitere de factură tradițională la un text presupunea eforturi de deplasare și căutare prin biblioteci, acum textul la care se face trimitere sau imaginea
5 apar cu mare repeziciune pe ecran.

B.1.3.5.1.1 Vim și navigarea prin intermediul *tag*-urilor

Dacă-ai lucrat cu Vim sau cu alte editoare pentru programare ai observat în meniuri sau la explicațiile pictogramelor că există posibilitatea de a construi *tag*-uri. Pe bara cu instrumente din Vim
10 vedeți o pictogramă cu un ciocan pus peste o etichetă de forma celor care se leagă de valize la aeroporturi. Le voi zice *tag*-uri, pentru a nu le confunda cu etichetele (*labels*).

Vim n-are inclus în *kit*-ul de instalare programul *ctags* care creează *tag*-uri. Acest program trebuie descărcat separat de la adresa
15 de Internet <<http://ctags.sourceforge.net>>. Programul *ctags* este scris de către Darren Hiebert. Acest program poate crea *tag*-uri pentru o sumedenie de limbaje de programare. Versiunea lui Hiebert nu include însă și L^AT_EX.

Programul lui Hiebert poate fi extins pentru a include și *tag*-
20 uri pentru L^AT_EX. Mergeți la adresa <<http://www.unb.ca/chem/ajit/vim.htm>> și veți găsi o versiune pentru Windows a *ctags* care produce și *tag*-uri pentru L^AT_EX. Sursele suplimentare necesare pentru a recompila programul lui Hiebert cu sprijin pentru L^AT_EX sunt scrise de către Ajit J. Thakkar.

tag-uri
pentru
L^AT_EX

25 Ce faceți cu *ctags*? Îl puneți într-un dosar aflat pe un path, o cale pe care caută sistemul Windows. Asta este tot. Vim are deja pregătite comenzile pentru crearea de *tag*-uri.

După ce ați creat *tag*-urile puteți să examinați fișierul *tags*, aflat în dosarul proiectului dumneavoastră. În proiectul cărții de față
30 fișierul respectiv are, de pildă, următorul rând:

```
1 ComenzileNoi anexe05tex.tex /\label{ComenzileNoi}$/" 1
```

Pe acest rând putem desluși mai întâi numele unei etichete, apoi cel al unui fișier și o expresie regulată. Litera 1 din finalul rândului ne spune că este vorba despre o etichetă (*label*). Expresia regulată permite Vim să caute eticheta cu numele respectiv.

35 Acum putem explica rațiunea pentru care am recomandat ca numele etichetelor să nu conțină două puncte. Dacă le conține, comanda de căutare generată de *ctags* nu va funcționa în Vim.⁴⁵

⁴⁵ Atenție, de asemenea, la respectarea diferenței dintre majuscule și minus-

Dacă dați, în Vim, un clic pe pictograma cu *tag*-ul galben, atunci când cursorul Vim este plasat pe numele unui *tag*, cursorul se va deplasa la rândul unde este declarată eticheta respectivă sau alt element cărui *ctags* i-a atașat un *tag*. În meniul **Tools** din Vim găsiți comanda necesară pentru a reveni la punctul de plecare.

verificarea
argumen-
telor
comenzii
`\ref{}`

Tag-urile sunt atât de utile încât am considerat necesar să modific bara de instrumente Vim în așa fel încât (în locul pictogramei pentru compilarea de surse) am pus o pictogramă pe care dacă dau un clic cursorul revine la punctul de plecare. Mișcările acestea sunt extrem de utile pentru a verifica dacă argumentul unei comenzi gen `\ref{}` este corect ales. Dacă greșesc numele etichetei, compilatorul emite un mesaj de eroare. Reparațiile în faza compilării sunt însă mult mai complicate.

Cu ajutorul *tag*-urilor puteți și naviga prin sursele \LaTeX . Puteți pune trimiteri și-n textul invizibil (la compilare) al sursei. De pildă, dacă puneți numele unei etichete într-un comentariu, puteți naviga de la acel punct la locul unde este declarată eticheta. Puneți doar numele, nu redeclarați eticheta. Puteți pune numele etichetei și imediat după comanda `%` prin care sunt introduse în sursă comentariile.

Programul *ctags* este foarte util și s-ar putea să vreți să-l extindeți. Eu, de exemplu, am adăugat în fișierul *latex.c* scris de Ajit J. Thakkar următoarele rânduri de cod:

```
1 addTagRegex (language,
2 "\\linelabel\{[ \t]*([~] \t)+[ \t]*\}",
3 "\\1", "1,linelabel", NULL);
```

Modificarea sursei programului este banală, dar folositoare. Doream să pot găsi și etichetele care sunt puse pentru a afla numărul rândului de pe pagina generată de către \LaTeX .

Compilarea sub Windows s-a dovedit a fi însă mai dificilă decât credeam. Nu știu exact cum au reușit Darren Hiebert și Anjit J. Thakkar să compileze sursele și să obțină un program DOS. Cert este că eu n-am reușit decât compilarea sub Cygwin. Dacă aveți emulatorul de Unix instalat, dați obișnuitele comenzi `./configure` și, apoi, `make`. Partea delicată este că executabilul *ctags* compilat sub Cygwin depinde tot timpul de fișierul *cygwin1.dll*.

ctags și
sort

Versiunea *ctags* compilată cu *cygwin* apelează programul *sort* de tip Unix! Trebuie să ștergeți cu totul programul *sort.exe* din `\Windows\command` sau să modificați căile pe care caută Windows

cule. \LaTeX și Vim își au rădăcinile în lumea Unix, iar „Unix nu-i ca Windows”. Unix este sensibil la diferența dintre „*eser*” și „*Eser*”.

în așa fel încât să găsească mai întâi pe `sort.exe` de tip Unix. Programul Unix `which` vă poate spune pe ce cale se află un executabil și, implicit, ce executabil va fi apelat cu prioritate.⁴⁶

Versiunea compilată cu Cygwin apelează și programul de tip `sh.exe`
 5 Unix `sh.exe`. Eu folosesc versiunea instalată de Msys pentru `sh.exe` și pe cea din GnuWin32 pentru `sort.exe`⁴⁷.

Pentru început versiunea programului `ctags` creată de către An-
 jitt J. Thakkar este absolut recomandabilă. Explicațiile de mai sus
 au fost doar pentru ucenicii vrăjitori.⁴⁸ Experiența descrisă arată,
 10 de altfel, limitele lucrului cu unelte Unix sub Windows. Dacă vreți
 să compilați programe C/C++ cu sursă deschisă, se pare că Linux
 este incomparabil mai bun. Atâta vreme cât n-aveți aceste interese
 de ucenic vrăjitor experimentat, Windows cred că rămâne o alegere
 mult mai rezonabilă.

15 **B.1.3.5.2 Comenzile noi** Ca la orice limbaj de programare,
 secretul producerii la nesfârșit de noi expresii rezidă și-n L^AT_EX în
 posibilitatea de a combina elementele deja introduse pentru a realiza
 noi construcții.

Ați remarcat desigur faptul că unele comenzi L^AT_EX au nume
 20 foarte lungi. Se spune, de exemplu, `\textbackslash{}`. Numele
 acesta lung nu este fără sens. El ne oferă deja o explicație a ros-
 tului comenzii; ne atrage, de asemenea, atenția asupra faptului că
 este o comandă pentru modul `text`. Dezavantajul este că e dificil de
 tastat. Nu este greu să definim o nouă comandă, cu un nume mai
 25 scurt.

Unii folosesc comenzile noi pentru a abrevia denumiri de organi-
 zații sau orice fragment de text ceva mai lung și a economisi timp
 la tastare. Comenzile noi se introduc după modelul următor:

```
1 \newcommand{\bs}{\textbackslash}
2 \newcommand{\ub}{Universitatea din București}
```

Comanda pentru construirea unei comenzi noi are două argumente:

⁴⁶Ordinea în care sunt listate căile în comanda `PATH` contează!

⁴⁷Programele GnuWin32 au prioritate în `path` față de cele Msys, iar Cygwin este instalată autonom, fără posibilitatea de apelare din linia de comandă MS-DOS. Doar fișierul `cygwin1.dll` este evident pus pe o cale unde poate fi găsit de către Windows.

⁴⁸S-ar putea ca unul dintre ucenici să descopere lesne cum se compilează sursele `ctags` fără Cygwin. Eu am încercat cu Mingw și cu compilatorul C/C++ oferit gratuit, pe Internet, de către firma Borland, dar n-am reușit. Fără discuție, nu se pune problema ca ucenicii să pirateze unul dintre compilatoarele comerciale. Acest lucru ar trebui exclus din principiu.

unul pentru numele noii comenzi; altul pentru descrierea, în \LaTeX , a acțiunii care este întreprinsă de către sistem în momentul în care este dată comanda respectivă. Efectul comenzilor noi de mai sus este lesne de ilustrat:

```
\bs{}NumeleComenzii\  
\ub
```

\backslash NumeleComenzii Universitatea din București
--

5

Comenzile noi (cel puțin pentru versiunea limbajului \LaTeX inclusă în distribuția MikTeX 2.4) pot fi definite oriunde în document. Recomandarea noastră ar fi să nu le plasați decât în antetul documentului principal sau într-un fișier special⁴⁹.

O altă recomandare ar fi aceea de a nu abuza de prescurtări gen `bs` sau `ub`. Ele scad gradul de lizibilitate al sursei. Putem foarte lesne folosi meniuri Vim pentru a insera în text cu un singur clic texte lungi care se repetă frecvent.

10

B.1.3.5.2.1 Comenzi noi cu argumente Cum procedăm însă atunci când vrem ca noua comandă să aibă și ea o parte sau mai multe părți variabile, unul sau mai multe argumente? Să zicem că vrem o comandă pentru a tipări comenzile \LaTeX sub forma unui text. În orice caz ne trebuie o parte variabilă pentru numele comenzii.

15

S-ar putea ca multă lume să știe din logică ideea de a compara o variabilă cu o lacună într-un text. În \LaTeX asemenea lacune sunt notate cu simbolul `#`. În mod obligatoriu acest simbol trebuie să fie urmat de un număr. Aici numărul este ca și un nume pentru lacuna respectivă.

20

Acum s-a dezlegat și misterul ultimului simbol rezervat. Simbolul `#` servește deci la marcarea, în definiția unei noi comenzi, a locului argumentelor.

25

Să vedem cum ar arăta practic comanda noastră pentru transformarea comenzilor \LaTeX în text:

```
1 \newcommand{\cmd}[1]{\texttt{\bs{#1}\{}}}
```

Se observă că, după numele comenzii noi, urmează o paranteză dreaptă în care este specificat numărul de argumente. Noi avem nevoie de un singur argument, pentru numele comenzii. Exemplul care urmează ilustrează folosirea noii comenzi.

30


```
\cmd{section}\\
\cmd{label}\\
\cmd{ref}
```

```
\section{}
\label{}
\ref{}
```

B.1.3.5.3 Mediile noi Putem extinde nu numai mulțimea comenzilor L^AT_EX, ci și pe aceea a mediilor L^AT_EX. Exemplul nostru este, din nou, foarte simplu. El ilustrează însă principiul general

5 folosit pentru a construi medii noi.

În unele cărți există pasaje scrise cu litere mai mici. Aceste pasaje discută uneori detalii ale unei probleme. Alteori sunt aprofundate aspecte mai dificile ale chestiunii discutate. Pentru a realiza un mediu în care scriem asemenea pasaje putem folosi următoarea

10 definiție:

```
1 \newenvironment{caNotaDeSubsol}{\footnotesize }{\normalsize}
```

Comanda prin care definim un mediu nou are trei argumente: numele mediului, o specificare a ceea ce se întâmplă când începe mediul și o specificare a ceea ce se petrece când mediul se încheie.⁵⁰

Să folosim acum mediul nostru:

```
Text normal.\\
\begin{caNotaDeSubsol}
Text care poate fi omis
la o primă lectură.
\end{caNotaDeSubsol}\\
15 Din nou text normal.
```

```
Text normal.
Text care poate fi omis la o primă lec-
tură.
Din nou text normal.
```

Se observă că mediul face exact ce i-am spus să facă. De pildă, la ieșire, nu rupe rândul, nu creează nici aliniat; doar trece la o literă cu corpul normal în textul respectiv.

Ca și comenzile, mediile pot fi definite în orice punct al textului.

20 Lucrul acesta poate să aibă sens în cazul exemplului de mai sus, pentru că nu folosesc acest mediu decât în această secțiune. Nu sens să procedăm așa când vrem să folosim sistematic un mediu.⁵¹

⁴⁹Pentru fișierul special, care are extensia *sty* a se vedea §B.1.3.6.2

⁵⁰Termenii *begin* și *end* indică tocmai ceea ce arată și sensul lor în engleză: un început și un sfârșit.

⁵¹Programul *ctags* (vezi aici §B.1.3.5.1.1) este de mare ajutor în reperarea definițiilor comenzilor și mediilor. Cu tot sprijinul pe care l-am primi din partea *tag*-urilor în regăsirea definițiilor, cred că este absolut recomandabil să construim în mod disciplinat sursa L^AT_EX.

B.1.3.6 Ziua a șasea

\LaTeX ne oferă toate avantajele unui limbaj de programare. Desigur, el este prea specializat pe tipărirea de texte pentru a-l folosi comod în alte scopuri.⁵² Când este vorba despre tipărirea de texte avantajul \TeX și \LaTeX este acela de a ne permite să punem textul în pagină așa cum dorim.

5

B.1.3.6.1 Pachetele \LaTeX S-ar putea ca multe dintre problemele care ne frământă pe noi să fi fost rezolvate de multă vreme de către alții. Comunitatea persoanelor care folosesc \LaTeX este puternică și aproape sigur veți găsi pe cineva care a dat deja o soluție.

10

Putem desigur îmbunătăți rezolvările date de către alții. Le putem mula după dorințele noastre. n-are însă rost să „reinventăm roata“.

Rezultatele obținute deja sunt „împachetate“ și doar trebuie instalate, integrate în sistemul \LaTeX pe care-l folosiți. Mik \TeX instalează o mulțime de astfel de pachete.

15

Sistemul trebuie anunțat când vreți să folosiți un anumit pachet. Persoanele care știu un limbaj de programare gen Pascal sau C/C++ sunt probabil intrigate de faptul c-am tot insistat asupra calităților ce limbaj de programare ale \TeX / \LaTeX . Unde sunt structurile condiționale? Puneți în antet⁵³ comanda următoare:

20

```
1 \usepackage{ifthen}
```

Comanda cere sistemului să utilizeze pachetul `ifthen`.

contoare Sistemul are o serie de contoare în care ține date importante pentru așezarea în pagină. Așa este, de pildă, contorul `page`, unde este ținut numărul paginii curente.

25

⁵²Leslie Lamport vorbește tot timpul despre \LaTeX ca despre un sistem de *macro*-uri \TeX . Este o doză de modestie exagerată în acest fel de a vorbi. Pe de altă parte, dacă ținem cont că *macro*-urile ne permit definirea altor *macro*-uri, vedem că este vorba de un limbaj. În sens strict, un *macro* este o serie de instrucțiuni care pot fi apelate apăsând o tastă sau folosind un cuvânt-cheie. Procesoare de cuvinte din complexele de programe pentru birou dau posibilitatea utilizatorilor de a crea *macro*-uri. Termenul „macro“ sugerează ideea că ne aflăm la un nivel superior celui *micro*, unde operează limbajul de programare propriu-zis. Dacă tratăm \LaTeX în același fel, atunci limbajul propriu-zis este \TeX . Noi preferăm să tratăm \LaTeX ca pe un limbaj de nivel înalt și \TeX ca limbajul procesorului de texte.

⁵³Dacă puneți această comandă în corpul documentului, veți primi un mesaj de eroare de la compilator.

Putem chiar afișa numărul paginii scriind `\thepage{}` în text. Prefixul `the` se pune în fața oricărui contor, atunci când vrem să-i aflăm valoarea. Putem afla valoarea din contor cu ajutorului lui `\value{page}`, dar această comandă o putem folosi doar în argumentul unei alte comenzi, cum ar fi `\isodd{}`, care ne spune dacă un număr este impar sau nu.

Acum puteți scrie o comandă al cărei rezultat ne va spune dacă suntem pe o pagină pară sau pe una impară:

```
\ifthenelse{\isodd{\thepage}}
{Pagina este impară!}
{Pagina este pară!}
```

Pagina este impară!

Comanda aceasta, cu unele îmbunătățiri,⁵⁴ vă poate fi de real folos dacă vreți ca aspectul unei pagini pare să difere de cel al unei pagini impare. De exemplu, s-ar putea să vrem să plasăm diferit o imagine. Nu trebuie să-mi fac griji dacă modific documentul. Structura condițională va asigura modificarea aspectului paginii.

Comanda `\ifthenelse{}{}{}` are trei argumente: o condiție, un argument care este executat atunci când condiția este adevărată și unul care este executat când ea este falsă.

Unde se află, de fapt, `ifthen`? Cum l-am putea găsi? Căutați în dosarul unde este instalată distribuția MikTeX un fișier denumit `ifthen.sty`. Îl găsiți pe o cale `tex\latex\base`. Face parte din nucleul sistemului L^AT_EX. Acest nucleu, în întregime sa, este tratat, la instalare, ca un pachet unic, denumit `ltxbase`. Nu puteți însă apela direct acest pachet. Trebuie să apelați componentele sale.

Cum putem afla ce pachete suplimentare a instalat sau poate instala MikTeX? În versiunile 2.3 și 2.4 folosiți MikTeX Package Manager. Cu ajutorul acestui program puteți găsi informații despre pachete.

Trebuie să treacă însă ceva timp până puteți folosi pachete ceva mai complicate. În orice caz trebuie citită documentația aferentă și testat totul pe fișiere destinate exercițiilor, înainte de a trece la utilizarea propriu-zisă.

Nu modificați în nici un caz fișierul `ifthen`. Dacă simțiți nevoia

⁵⁴Rezultatul comenzii `\thepage` depinde de generarea numărului paginii; în partea de sus a paginii veți găsi în `page` încă numărul paginii anterioare (a se vedea explicația în L^Amport[21, p.135]). Soluția sugerată de către L^Amport[21, p.196] este să declarăm o etichetă de genul `aceastaPagina` și apoi să dăm condiției forma `\isodd{\pageref{aceastaPagina}}`. Atenție și la necesitatea de a da mai multe compilări pentru a obține rezultatul dorit. Citiți atent avertismentele din mesajele compilatorului!

să experimentați, faceți o copie a fișierului, redenumiți-o și puneți-o în dosarul `localtexmf`⁵⁵.

Nu aș sfătui totuși pe nimeni să experimenteze cu `ifthen`. Un exemplu mai practic este dat în secțiunea care urmează.

B.1.3.6.1.1 Letrinele Unora le place să înceapă secțiunile mai importante, cum ar fi capitolele unei cărți, cu o literă supradimensionată, care are înălțimea a două-trei rânduri. Această literă se numește *letrină*.

Am putea meșteri, urmând indicațiile din literatură,⁵⁶ o comandă pentru letrine:

```
1 \newcommand{\letrina}[3]{\noindent\hangindent=#1
2 \hangafter=-#2\hskip-#1\smash{\hbox to #1
3 {\#3\hfill}}\ignorespaces}
```

```
\letrina{15pt}{2}
{\Huge E}seul filosofic
```

Eseul filosofic

Exemplul acesta este dat special pentru a arăta ce util ar fi un pachet special dedicat rezolvării problemei cu care ne confruntăm.

Pentru exercițiul real încărcăți pachetul `letrine`.

```
\letrine[lines=1]{E}seul
filosofic
```

Eseul filosofic

Soluția lui Daniel Flipo, autorul pachetului `letrine`, este evident mai ușor de folosit. Dacă studiați documentația aferentă pachetului, veți descoperi o mulțime de posibilități de a crea letrine.

B.1.3.6.2 Fișierul de tip `sty` Comenzile de utilizare a pachetelor, definițiile de noi comenzi și de noi medii sunt reutilizabile. Dacă le-am pune într-un fișier, am putea muta lesne fișierul respectiv dintr-un proiect într-altul. În plus, fișierul respectiv ar face ca antetul documentului principal să fie mult mai concis.

Extensia fișierului în care punem pachetele pe care le utilizăm și definițiile comenzilor și mediilor este `sty`.

Sintaxa fișierelor de tip `sty` diferă de cea a din fișierele de tip `tex`. Diferențele față de ceea ce am văzut până acum sunt ușor de sesizat, dacă cercetați cu atenție exemplul de mai jos. Am extras chiar o parte din fișierul `sty` folosit pentru această carte.

⁵⁵ Acesta este, de regulă, numele dosarului unde stau fișierele create local sau pachetele instalate suplimentar de către dumneavoastră. Nu uitați să folosiți programul `MikTeX Options` pentru a reimprospăta baza de date a sistemului `TeX`.

⁵⁶ Aici folosim ideile lui Seroull[39, p.83]

```

1 \ProvidesPackage{eseu}
2 \RequirePackage{ifthen}
3 \RequirePackage{lettrine}
4 \RequirePackage{url}
5 %=====
6 \setcounter{secnumdepth}{5}
7 %=====
8 \newcommand{\bs}{\textbackslash}

```

Pe primul rând se arată modul în care se comunică L^AT_EX care este pachetul furnizat. Apoi se vede cum, în loc de `\usepackage{}` trebuie folosită comanda `\RequirePackage{}`

În fișierul `sty` putem defini comenzi sau medii noi. Putem, de asemenea, modifica elemente ale sistemului: valori ale unor variabile, definiții ale unor comenzi și medii. Pe rândul 6 din exemplul de mai sus este modificată valoarea până la care L^AT_EX dă numere secțiunilor unei cărți.

Dacă redefiniți o comandă trebuie să faceți acest lucru în mod explicit: folosiți `\renewcommand{}`. Dacă redefiniți un mediu, folosiți `\renewenvironment{}`.

Utilizarea fișierului `sty` creat este foarte simplă. În cazul fișierului `eseu.sty`, în fișierul principal al proiectului introducem o comandă `\usepackage{eseu}`. Acest mod de a proceda este foarte logic; la urma urmei, fișierul `sty` creat constituie pachetul cu soluțiile specifice proiectului respectiv.

B.1.3.7 A șaptea zi

Dacă-ai ajuns până aici, nu doar cu lectura, ci și cu exercițiile, este foarte bine. Aproape c-am putea da comanda T_EX `\relax`. Această comandă este dată atunci când argumentul trebuie să fie o acțiune, dar – în fapt – sistemul n-are nimic de întreprins.⁵⁷

Oricine utilizează un procesor de cuvinte obișnuit, la nivelul cerut de un eseu filosofic curent (fără tabele, formule, imagini), poate să obțină acum cu L^AT_EX tot ce ar obține și cu editorul de birou. Cu unele excepții! Ce facem dacă vrem să schimbăm structura paginii, de pildă? Acest lucru este ușor de făcut cu editorul de birou.

⁵⁷Seroul[39, p.288] explică utilitatea deosebită a acestei comenzi. Fără a intra în detalii, putem spune că ea servește la eliminarea unor ambiguități care ar putea fi generate prin combinarea comenzilor T_EX și a textului documentului. Sistemul trebuie să știe unde să se oprească. În documentele scrise în L^AT_EX nu veți vedea această comandă. Dacă vă uitați prin fișiere `sty` sofisticate aveți destul de repede șansa să vă întâlniți cu ea.

Răspunsul la întrebarea de mai sus îl vom da în continuare. Ar trebui precizat însă că, mai important, în acest moment, este să treceți în revistă întreg complexul de resurse oferit de către \LaTeX .

B.1.3.7.1 Structura paginii Dacă scrieți o lucrare de seminar sau o lucrare de diplomă, recomandarea noastră fermă este să nu modificați dimensiunile standard furnizate automat de către \LaTeX . Sistemul este special conceput pentru a ne bate cât mai puțin capul cu forma textului. Autoarea sau autorul răspund de conținut, \LaTeX se îngrijește de formă.

Ce ne facem însă dacă intră în joc alte constrângeri? O carte are, de exemplu, un format cu particularitățile sale. În acest caz, este acceptabilă modificarea structurii paginii.

\LaTeX n-ar fi deloc un limbaj comod dacă am vrea să scriem programe care să rezolve ecuații. Nu aceasta este menirea sa. Formatul comenzilor, numele lor ample și explicite sunt însă extrem de utile pentru tehnoredactare. Rândurile de mai jos sunt extrase din fișierul `sty` al cărții de față. Ele aproape nu necesită comentarii.

```
1 \oddsidemargin=2.2mm
2 \evensidemargin=10mm
3 \topmargin=0mm
4 %\headwidth %vezi mai jos
5 \headheight=5.5mm
6 \headsep=4mm
7 \textheight=200mm
8 \textwidth=112mm
9 \marginparsep=3mm
10 \marginparwidth=20mm
11 \footskip=5.5mm
12 \marginparpush=3mm
13 \hoffset=17.5mm
14 \voffset=12mm
15 \paperwidth=210mm
16 \paperheight=295mm
17 %=====
18 \addtolength{\headwidth}{\marginparsep}
19 \addtolength{\headwidth}{\marginparwidth}
```

Ultimele două comenzi măresc lățimea antetului paginii, pentru a acoperi și coloana destinată notelor marginale. Altfel, `headwidth` coincide cu lățimea textului.

B.1.3.7.2 Ce mai știe \LaTeX ? \LaTeX lucrează cu cutii (*boxes*) pe care le așează pe orizontală și verticală. Dacă umplem o cutie cu

cerneală și așezăm lângă ea alte cutii și așa mai departe ar ieși un desen. L^AT_EX poate executa desene în acest mod.

Recomandarea noastră ar fi să folosiți L^AT_EX doar pentru a trasa linii și forme geometrice elementare. Partea forte a L^AT_EX o constituie aranjarea textului în pagină, nu desenarea de figuri. Figurile pot fi create cu alte programe și apoi pot fi integrate în text.

Mari probleme creează desigur tipurile de litere, tabelele, imaginile. L^AT_EX are soluții, pe care le prezentăm pe scurt în anexe separate.

Pe lângă compilatorul de L^AT_EX ca atare, orice distribuție L^AT_EX include două programe absolut formidabile: un program pentru lucrul cu bibliografia și unul pentru crearea de indici. Împreună cu aceste programe L^AT_EX vă permite să creați lucrări care au cu adevărat forma unor lucrări academice. Și aceste programe sunt prezentate aici în anexe separate.

Pe lângă programe, există în distribuția MikT_EX și un alt element prețios: documentația. Pentru început, identificați modul în care puteți accesa documentația și răsfoiți câteva titluri.

Căutați în dosarul MikT_EX calea `texmf/doc/guides`. Veți găsi acolo un dosar `lshort-english`. În el se află o carte scrisă de Tobias Oetiker, Hubert Partl, Irene Hyna și Elisabeth Schlegl *The Not So Short Introduction to L^AT_EX2_ε*. Este o introducere foarte bine făcută și puteți să o folosiți cu mare succes pentru a consolida cunoștințele de bază în materie de L^AT_EX. Documentația include versiuni ale acestei cărți în mai multe limbi.

documentația
MikT_EX

Sursele cărții lui Oetiker et al. se află în dosarul `\texmf\source\lshort-english`. Dezarhivați-le și creați, cu ajutorul T_EXnicCenter un proiect având fișierul `lshort.tex` ca fișier principal. Dacă aveți o instalație completă, după două-trei compilări, obțineți un fișier de tip `dvi` acceptabil. Cartea este concepută pentru a fi transformată în `pdf`. Prefer `dvi`-ul în combinație cu T_EXnicCenter pentru că astfel pot fi studiate în condiții optime sursele. Vizualizați `dvi`-ul; dați un dublu clic în punctul care vă interesează și studiați sursa.

Sunt o mulțime de elemente interesante în documentație. Ne limităm aici doar la două recomandări suplimentare. Restul puteți descoperi și singure sau singuri.

Citiți articolul lui Guido Gonzato „L^AT_EX for Word Processor Users” din dosarul `latex4wp`. Veți înțelege cum puteți produce cu ajutorul L^AT_EX tot ceea ce puteți realiza cu ajutorul unui procesor de cuvinte. Cu alte cuvinte, aflați de ce n-aveți nici un motiv să regretați trecerea de la editorul de birou la L^AT_EX.

Un articol scurt, dar extrem de interesant, este cel al lui Piet van Ostrum, „Page Layout in L^AT_EX“. Îl găsiți, în dosarul fancyhdr, pe calea `\texmf\doc\latex`. Articolul oferă explicații pertinente cu privire la structura paginii.

B.1.3.7.3 Forța cooperării dintre Vim, T_EXnicCenter și Yap 5
Să zicem c-ați compilat cartea scrisă de Oetiker et al. Ați găsit însă o comandă ciudată în surse. Deschideți fișierul respectiv în Vim. Creați fișierul `tags`. Căutați acum comanda respectivă.

Dacă vreți să aflați, de exemplu, cum a fost definit mediul `code`, puteți face acest lucru cu ajutorul *tag*-urilor. Definiția este, de altfel, uimitor de simplă: este vorba de folosirea unui mediu standard pentru citate. 10

F1 Dar dacă vreți să aflați ce sens are comanda `\verb`. n-are *tag*! Semn că este un element standard din L^AT_EX. Reveniți în T_EXnicCenter. Puneți cursorul pe cuvântul respectiv și apăsați tasta F1. 15
În câteva clipe vă este afișată într-o fereastră specială explicația comenzii.

Vim și cu T_EXnicCenter constituie o combinație foarte puternică. Aveți atât acces direct la surse, cât și deplasarea între sursă și programul de vizualizare (în ambele direcții). În Vim puteți crea pentru fiecare document sau gen de document la care lucrați meniuri adaptate cerințelor dumneavoastră. Puteți naviga prin surse cu ajutorul *tag*-urilor. Puteți ajunge în Vim și din programul de vizualizare.⁵⁸ 20

De ce n-am folosi totuși un program de tip WYSIWYG? Există atât programe gratuite, cât și programe comerciale pentru lucrul în L^AT_EX în acest mod (cel al editoarelor uzuale de birou). 25

Lyx este un program utilizat, de regulă, sub Linux pentru a edita în maniera WYSIWYG documente L^AT_EX. Lyx poate fi funcționa și sub Windows, cu ajutorul Cygwin. Trebuie să aveți instalat însă și emulatorul de X Window. X este interfața grafică tipică pentru sistemele Unix. 30

Forța combinației dintre un editor precum Vim și un mediu integrat pentru compilare așa cum este T_EXnicCenter ni se pare însă net superioară. Posibilitățile de lucru sunt mai bune. Separarea dintre conținutul și forma textului este clară. În plus, prelucrarea fișierelor în Vim este extrem de sigură. Orice sistem integrat are fragilitatea sa și, la lucrul cu fișierele mari, s-ar putea să vă producă surprize neplăcute. 35

⁵⁸ Vedeți explicațiile din §B.1.2.3.1.1.

B.2 BibT_EX

Dacă o persoană scrie un program de calculator, primul lucru la care se gândește este cum ar proceda „manual“. Dacă știi să rezolvi „cu mâna“ problema, atunci poți trece la faza în care extragi algoritmul, 5
ideea din spatele soluției, și o transferi către programul de calculator. Principiul care stă la baza înțelegerii elaborării și tehnoredactării computerizate a textelor este același.

B.2.1 Primii pași în lumea bazelor de date

Modul tradițional de alcătuire a unei bibliografii presupunea o colecție de fișe bibliografice și o listă alcătuită la finalul lucrării pe 10
baza acestor fișe. Elaborarea unei bibliografii computerizate merge pe aceeași idee.

Din punctul de vedere al programării lucrurile se complică însă. Până acum am avut de a face cu o sursă care poate ori să fie interpretată, în care caz se execută pas cu pas comenzile din sursă, ori 15
să fie compilată, în care caz sursa este transformată într-un fișier de alt tip.

Bibliografiile ne pun în fața unei situații diferite. Chiar din perspectiva tradițională, fișele bibliografice erau o *bază de date*. În baza 20
de date, pe fișe, câmpurile cu informații pot fi dispuse în cu totul alt mod decât în lista bibliografică. De asemenea, în listă putem prezenta doar o parte din informațiile din baza de date.

Prin urmare, în computer, ne vom confrunta cu o diversificare a fișierelor și a programelor care lucrează cu ele. Unele fișiere vor 25
stoca baza de date ca atare. Altele vor fi surse (programe) pe baza cărora vor fi extrase și prezentate date.

Din punctul de vedere al programelor care lucrează cu bazele de date, principiul tare al surselor deschise simplifică lucrurile. Fișierele care conțin datele sunt fișiere de tip text. Noi le vom prelucra cu 30
ajutorul Vim.

O fișă bibliografică tradițională are însă câmpuri! Într-un fișier de tip text nu putem însă trage linii pentru a separa câmpurile. Se folosesc, în schimb, separatorii. Separatorul este un semn cu ajutorul 35
cărui distingem câmpurile.

Un exemplu ne va lămuri lesne despre ce este vorba. Iau trei cărți dintre cele care sunt în bibliografia cărții de față și construiesc un fișier cu următoarele câmpuri: o prescurtare făcută din numele au- 35
toarei sau autorului, numărul total de pagini, numărul de pagini din **fișier csv**

partea introductivă a cărții⁵⁹, numărul de pagini din partea principală a cărții, numărul de pagini alocate indicelui. Ca-n orice înșiruire aceste date sunt separate prin virgulă. Fiecare rând corespunde unei fișe tradiționale:

¹ Bazerman 1989,527,14,501,12

² Fischel 1984,299,10,280,9

³ Hairston 1974,364,13,344,7

Un asemenea fișier se numește fișier de tip `csv`⁶⁰.

5

În funcție de interesele noastre, separatorul ar putea fi și altceva decât virgula: un spațiu alb, un semn mai deosebit. Extensia fișierului este însă întotdeauna `csv`.

Există o serie întreagă de operații care pot fi făcute cu un fișier `csv`. Fișierul de mai sus poate fi, de pildă, sortat în funcție de numărul total de pagini. Operația este absolut similară sortării fișelor tradiționale de hârtie.

10

Tot așa cum din baza tradițională de date putem extrage anumite informații, din fișierul de mai sus am putea construi liste cu numărul de pagini al părții introductive sau liste cu numărul de pagini al

15

Bazele de date pot fi punctul de plecare pentru calcule. Pornind de la informațiile de mai sus am putea, de exemplu, calcula procentul din total al numărului de pagini al indicelui. Folosind fișiere similare am putea face, pornind de la datele lor, calcule statistice mai complicate sau le-am putea folosi pentru a testa ipoteze statistice.

20

Un program simplu de operare cu fișiere `csv` este `CSVdb`, scris de Sam Francke.⁶¹ Programul este *cardware*⁶².

Este recomandabil să folosiți acest program sau un program similar pentru a vă face practic o idee despre fișierele `csv`. Cu programul lui Sam Francke puteți sorta, schimba ordinea coloanelor, elimina fișele⁶³ care se repetă. Este posibilă, de asemenea, utilizarea a diverși separatori.

25

⁵⁹Ceea ce în engleză se numește *frontmatter*.

⁶⁰De la sintagma englezească *comma separated values*.

⁶¹Pagina de web a programului este <http://home.hccnet.nl/s.j.francke/software/software.htm>

⁶²Dacă folosiți programul și vă place, atunci aveți obligația (morală) de a-i trimite autorului o vedere la adresa indicată în documentația programului.

⁶³În jargonul computerelor fișelor tradiționale li se spune „înregistrări”, de la englezescul *records*.

B.2.2 Sistemul BibTeX

Un fișier csv are însă o structură foarte incomodă din punctul de vedere al fișelor bibliografice. Fișele bibliografice sunt de diverse tipuri. Fiecare tip are câmpurile sale specifice. Câmpul pentru titlul revistei n-are sens la o carte, de exemplu.

Am putea lăsa necompletate câmpurile care nu au sens. Soluția nu este nici elegantă, nici logică. Logic ar fi să distingem explicit tipurile de fișe în formatul electronic.

Probabil că atunci când ați instalat T_EXnicCenter ați observat că acesta integrează și un compilator botezat BibTeX. Acesta este capabil, pe baza unor programe, să producă fișiere care pot fi utilizate de L^AT_EX pentru a genera liste bibliografice.

Pentru început, trebuie studiate fișierele care alcătuiesc baza de date propriu-zisă. Aceste fișiere au extensia bib. Vim are capacitatea de a le colora într-un mod care ne ajută să ne dăm seama dacă fișiere sunt construite corect sau nu.

În formatul bib tradițională fișă bibliografică a unei cărți are un aspect de genul următor:

```
1 @BOOK{latex,
2   author={Leslie Lamport},
3   title={\LaTeX}: a document preparation system},
4   address={Reading, Massachusetts},
5   publisher={Addison Wesley Longman, Inc.},
6   year={1994}
7 }
```

Structura de mai sus are forma @TIP{eticheta, câmpuri}. Câmpurile sunt separate prin virgule. Fiecare câmp are un nume, urmat de semnul egal, după care este pus conținutul propriu-zis al câmpului.

Rolul etichetelor este exact același ca și-n cazul unui argument al comenzii \label{} din L^AT_EX sau al unui tag. Eticheta unei înregistrări ne permite să regăsim înregistrarea respectivă într-un fișier.

Numele câmpurilor sunt suficient de sugestive pentru a nu necesita explicații. În cazul cărții date drept exemplu aceste câmpuri asigură informația bibliografică minimă în sensul specificat aici.

Pentru a facilita crearea de fișiere bib putem extinde meniurile Vim. Iată un exemplu pentru o intrare de tip BOOK:

```
1 :imenu Bib.&Book @BOOK{,<CR>author={},<CR>title={},
2 <CR>address={},<CR>publisher={},<CR>year={},
3 <CR><Home><Esc>6<Up><End>i
```

Articolele sunt adesea prezente în bibliografiile eseurilor filosofice. Prin urmare, este important să oferim și aici un exemplu de fișă bibliografică electronică pentru un articol. Această fișă ia forma unei înregistrări într-un fișier bib, ca-n exemplul care urmează:

```
1 @ARTICLE{FlewTCE,  
2   author={Antony Flew},  
3   title={Issues in Teaching Contemporary Ethics},  
4   journal={Teaching Philosophy},  
5   month={Summer},  
6   year={1975},  
7   volume={1},  
8   number={1},  
9   pages={55--60},  
10 }
```

Cum este și normal, în cazul unui articol, trebuie menționată publi-
cația periodică cu datele aferente.

Spațiul nu ne-ar permite să descriem pe larg diverse tipuri de
intrări într-un fișier bib. Din fericire, principiul sursei deschise oferă
o excelentă posibilitate de documentare. Dacă mergeți la dosarul
unde este instalat sistemul MikTeX, veți găsi în `\texmf\bibtex\bib`
o mulțime de exemple de fișiere bib.

În cazul în care nu există o descriere standard, trebuie să scriem
una. Pentru documentele de pe Internet am construit structura de
mai jos. Ea nu poate fi utilizată decât împreună cu `simply.bst`, o
versiune pentru limba română a lui `plain.bst`. Fișierul `simply.bst`
este creat special pentru această carte.⁶⁴

```
1 @WEBBED{,  
2   author={},  
3   title={},  
4   notatxt={},  
5   url={},  
6   year={},  
7   download={},  
8   note={},  
9   annote={},  
10 }
```

Câmpurile clasice sunt primele două. Ca orice document, un text
aflat pe Internet a fost scris de către cineva sau o organizație și-a
asumat răspunderea pentru el. De asemenea, are un titlu. Dacă
aceste elemente lipsesc, nu cred că are sens să figureze în bibliografia

⁶⁴ Pentru `simply.bst` v. situl cărții.

unui eseu filosofic. Ar putea fi cel mult menționat în note, cu titlu de sursă a unui exemplu sau de sursă a unei opinii sau afirmații.

În câmpul `notatxt` punem o notă despre situl pe care se află textul sau despre caracterul textului.⁶⁵ Conținutul acestui câmp fiind
 5 mai flexibil putem să-l reglăm în așa fel încât să scăpăm de cutiile cu litere care depășesc zona alocată textului bibliografiei (*bad boxes*).

În câmpul `url` trebuie pusă adresa de Internet. Acesta este un câmp absolut necesar, dar nu suficient pentru o astfel de fișă. Menționarea anului în care documentul a fost publicat pe Internet ar fi
 10 o completare utilă.

Documentele de pe Internet au o natură dinamică. Se schimbă des. De multe ori este greu să fie identificată versiunea. Nu de puține ori însă, după ce au fost disponibile o vreme, dispar. De aici necesitatea câmpului `download`, în care trebuie trecută data la care
 15 a fost descărcat⁶⁶ documentul. Recomandarea noastră ar fi să fie folosit un format al datei acceptabil pentru L^AT_EX și care nu recurge la numele de luni, pentru a nu crea confuzii cu formatul folosit în bibliografie pentru data publicării documentului. Un astfel de format ar fi, de pildă, implicit în: 14/09/2003.

În sfârșit, structura de mai sus este prevăzută cu un câmp pentru
 20 o notă și unul pentru adnotare. Nota poate să fie folosită pentru un supliment de informații privind documentul. Adnotarea are rolul uzual, acela de a oferi o scurtă descriere a documentului.

B.2.2.1 Utilizarea de sine stătătoare a BibT_EX

25 Un fișier bib poate fi utilizat și independent de sistemul L^AT_EX. Soluția care mi se pare cea mai bună, sub Windows, este cea a utilizării programului BibDB, creat de către Eyal Doron.⁶⁷

Chiar și o bibliografie cu peste 16000 de înregistrări, cum este cea creată de către Richmond Thomason pentru lucrările de filoso-

⁶⁵ Aceste informații le-am reținut pe descrierea din fișa bibliografică (v. pagina 16). Pentru folosirea în citări de sine stătătoare a acestei note v. pagina 75.

⁶⁶ Nu uitați că și atunci când aveți sentimentul că doar vedeți documentul, el tot este descărcat temporar.

⁶⁷ <<http://www.tcissoft.com/tcissoft/bibdb.html>> este pagina de web a programului BibDB. BibDB este un program cu surse deschise. Limbajul surselor este Pascal.

fia limbajului, semantică, inteligență artificială și domenii conexe,⁶⁸ este manevrată cu mare ușurință de către BibDB.⁶⁹

Folosirea interfeței grafice a BibDB este relativ facilă. Dacă v-ați însușit și utilizarea expresiilor regulate, atunci căutarea în baza de date va decurge foarte ușor.

5

B.2.3 Stilurile bibliografice

Datele dintr-un fișier bib pot fi extrase și prezentate în diverse moduri, indiferent de felul sau ordinea în care apar ele în fișierul respectiv. Modurile acestea de prezentare a datelor bibliografice se numesc „stiluri bibliografice“.

10

Descrierile stilurilor bibliografice sunt plasate în fișiere care au extensia `bst`. Ele sunt scrise într-un limbaj special, inventat de către creatorul programului `BIBTeX`, Oren Patashnik. Dacă nu știți ceva mai multă programare sunt puține șanse să înțelegeți limbajul fișierelor `bst`. În orice caz, nu modificați conținutul fișierelor `bst` standard! Dacă vreți să operați modificări, faceți o copie într-un dosar din `localtexmf` și redenumiți copia, în așa fel încât să nu se producă nici un fel de confuzii în sistem.

15

fișierele Pentru început puteți investiga fișierele `bst` cele mai obișnuite
bst din dosarul `base` plasat pe calea `\texmf\bibtex\bst`. Repet, nu modificați nimic aici! Faceți doar copia în modul indicat mai sus și examinați copia.

20

Fișierul cu care este bine să începeți este `plain.bst`. Căutați, de pildă, secvența `FUNCTION {book}` și cercetați definiția modului de afișare a datelor bibliografice pentru o carte.

25

Limbajul `bst` are o sintaxă aparte. Dacă în `LaTeX` ați văzut o structură condițională de forma „comandă, condiție, acțiune, acțiune“, în `bst` structura condițională are forma „condiție, acțiune, acțiune, if\$“. Este ca și cum ați scrie „2 2 +“. Există motive temeinice pentru a proceda așa, dar explicarea lor nu-și are rostul aici.⁷⁰

30

Fiecare stil bibliografic are drept efect moduri diferite de sepa-

⁶⁸ Căutați fișierul `rht.bib` la adresa de `<http://www.csse.monash.edu.au/mirrors/bibliography/Ai/rht.html>`. Dacă nu găsiți acolo fișierul `rht.bib`, încercați o căutare cu ajutorul cuvintelor-cheie.

⁶⁹ Am reușit să încarc chiar bibliografii mai ample de 43193 intrări și de 64307 intrări, în mai puțin de un minut, respectiv în jur de un minut. Testul l-am făcut cu un fișier obținut prin concatenarea unei colecții de fișiere `bib`. În orice caz, nu aș folosi pentru biblioteca de acasă altceva decât fișiere `bib`. Putem, de asemenea, medita și la ideea că o bibliotecă obișnuită, precum cea a unei facultăți, are cam 50000 de volume.

⁷⁰ Puteți citi explicațiile lui Oren Patashnik în `btzhak` din documentația

rare a elementelor intrării bibliografice, precum și unele diferențe în modul de evidențiere a lor. Ordinea faptelor privitoare la publicare diferă și ea. Dacă vă familiarizați cu funcțiile din `plain.bst`, veți descoperi desigur că acesta pune mai întâi editura și apoi localitatea și anul. Majoritatea stilurilor bibliografice preferă să indice întâi localitatea și apoi editura.

Stilurile bibliografice definite în fișierele `bst` sunt replici electronice ale stilurilor tradiționale. Un element important de diferențiere îl reprezintă modul în care se realizează conexiunea dintre text și intrările din bibliografie. Multe stiluri folosesc un cuplu „autor-an”. `Plain` folosește un număr, numărul intrării bibliografice.

Oren Patashnik argumentează în favoarea stilului `plain` contra sistemului editurii Universității din Chicago.⁷¹ Stilul acesta era dificil de folosit pe vremea tiparului tradițional. Cu `BibTeX` și `LaTeX` situația se schimbă. Nu trebuie să avem grija numerelor din bibliografie.

Pentru cartea de față am adaptat `plain.bst` la cerințele limbii române. Am făcut și unele modificări, în funcție de practica editurilor din România în domeniul bibliografiilor.⁷²

B.2.3.1 Listarea unei bibliografii în `LaTeX`

Generarea bibliografiei în `LaTeX` este foarte simplă. În fereastra de dialog cu proprietățile proiectului `TEXnicCenter` bifați caseta `Uses BibTeX`. În fișierul principal puneți în punctul în care trebuie să apară bibliografia rânduri de program `LaTeX` după modelul următor:

```
1 \backmatter
2 \nocite{*}
3 \bibliography{eseu}
4 \bibliographystyle{plain}
```

Comanda de pe rândul 1 spune `LaTeX` că este vorba despre partea finală a cărții. Comanda de pe rândul 2 cere afișarea întregului conținut al fișierului `bib`, indiferent de citarea sau nu a intrărilor în cuprinsul lucrării. Argumentul comenzii de pe linia 3 este numele fișierului `bib`, fără extensie. `BibTeX` adaugă automat extensia.⁷³

`MikTeX`. Țineți însă cont că acesta este un text pentru ucenicii vrăjitori, nu pentru utilizatorii obișnuiți!

⁷¹A se vedea documentul `btldoc` din documentația `MikTeX`.

⁷²Nu prezentăm aici fișierul `plainro.bst`. A se vedea pe situl cărții o prezentare a acestui fișier.

⁷³Dacă aș scrie `eseu.bib`, atunci `BibTeX` ar căuta un fișier `eseu.bib.bib`. Acest fișier nu există și bibliografia nu poate fi generată. Atenție și la faptul că,

Argumentul comenzii de pe rândul 4 este numele fișierului de tip bst care conține programul de prelucrare a bazei de date. Din nou, extensia NU trebuie inclusă în argument.

Recomandarea cât se poate de categorică pentru începători este să folosească mediul integrat T_EXnicCenter pentru a genera bibliografia. B_IB_TE_X este un compilator. Apelarea sa în linie de comandă cere o cunoaștere destul de bună a funcționării sale.

B.2.3.2 Trimiterile la bibliografie în L^AT_EX

După cum am explicat deja,⁷⁴ etichetele înregistrărilor din fișierele bib sunt folosite ca argumente pentru trimiteri. Comanda pentru trimiteri este `\cite{}`. În argumentul ei este pusă eticheta înregistrării la care vrem să trimitem.

Comanda `\cite{}`, dacă folosim stilul plain, generează un număr pus în paranteze drepte. Acesta este numărul intrării din bibliografie la care vrem să trimitem.

Forma `\cite[]{}` a comenzii pentru trimiteri la bibliografie include și o opțiune. Argumentul opțional recomandat este un număr de pagină, precedat de „p.“, în cazul limbii române. Am putea pune și un alt argument opțional, cum ar fi un număr de secțiune. Cartea sau articolul la care trimitem trebuie să aibă însă secțiunile numerotate. În cazul documentelor de pe Internet, putem indica un fișier.

Comanda `\cite{}` are diverse versiuni, în funcție de stilul bibliografic folosit. Aceste versiuni sunt descrise în documentația aferentă stilului bibliografic respectiv. Folosirea versiunilor comenzii scade însă portabilitatea documentului de la un stil bibliografic la altul.

B.3 Turnul Babel

Din punctul de vedere al calităților tipografice ale textului, departe de a fi o binecuvântare, calculatorul este adesea un adevărat blestem. Cea mai vie ilustrare o constituie modul în care sunt tratate literele românești. Nenumărate eseuri studentești sunt scrise fără diacritice. Ghilimelele tradiționale românești tind să dispară. Într-un cuvânt, calculatorul pare a ne întoarce într-o epocă de sălbăticie intelectuală.

Motivele acestei stări de lucruri cred că se reduc, până la urmă,

în L^AT_EX, fără a folosi pachete speciale, numărul de înregistrări din fișierul bib nu poate trece de 3000. Nu văd însă de ce un eseu filosofic ar avea nevoie de un fișier atât de mare.

⁷⁴A se vedea mai sus pagina 235, rândul 23.

la ignorarea importanței programării unui calculator. Consumatorii tind să achiziționeze „mașini puternice”. Programele sunt însă adesea piratate sau instalate fără o reglare atentă. Efectul este, fără îndoială, dezastruos.

5 B.3.1 Literele românești

Ignorarea importanței programării începe probabil de la felul în care este percepută tastatura. Simbolurile scrise pe taste au o valoare pur orientativă. Totul poate fi schimbat prin programare. Evident, nu trebuie neapărat să *scrieți* programele respective. Ajunge să-i
10 spuneți sistemului de operare ce program vreți să utilizați.

Să zicem că folosiți Windows98, sistemul de operare la care ne referim în mod preponderent în această carte. Dați un clic pe butonul Start. Urmați ruta Settings→ Control Panel→ Keyboard. Fereastra
care se deschide vă permite să controlați proprietățile tastaturii. **litere românești în Windows**
15 Alegeți panoul Language și dați un clic pe butonul Add. . . . Selectați din lista derulantă Romanian. Dați un clic pe OK. Apoi dați un clic pe Apply.

După ce-ați făcut operațiile de mai sus sistemul vă va cere, probabil, CD-ul de pe care ați instalat Windows98. Lucrurile se vor pe-
trece ceva mai lin dacă sistemul este preinstalat. De cele mai multe
20 ori însă, oricum nu este nevoie de CD. Fișierul de care aveți realmente nevoie este KBDRO.KBD și se găsește poate deja în dosarul c:\windows\system.

Verificați, după aceea, dacă pe *taskbar* va apare indicatorul ce
25 ne spune care este tastatura pe care o folosim. De asemenea, vedeți care este combinația de taste care vă convine cel mai mult atunci când comutați de la o tastatură la alta.

Acum aveți o tastatură care include literele românești. Fișierul KBDRO.KBD de pe CD-ul de instalare produce o tastatură pe care
30 eu unul nu o găsesc comodă. Evident, este mai bună decât una fără litere românești. Din păcate, tastele Y și Z își inversează în mod inutil rolul. Semnele de punctuație sunt greu de găsit. Dacă dați peste un suflet caritabil în stare să modifice în mod competent fișierul KBDRO.KBD, ar fi bine să aduceți tastatura la o formă mai
35 comodă.

La urmă, dar nu în cele din urmă în ordinea importanței, nu uitați că trebuie să alegeți un tip de literă care conține litere românești. Ca
exercițiu, mergeți în Vim la meniul Edit. De acolo la Select Font. . . și
testați existența la rubrica Script a opțiunii Central European; dacă **Vim și literele românești**

această opțiune nu există, atunci n-aveți litere românești în cazul tipului respectiv de literă.

Ideea de mai sus este foarte simplă. Este ca și cum i-am spune unui tipograf că vrem literele românești din cutare tip de literă. Nu ajunge să-i spunem acest lucru. El trebuie să și găsească literele respective prin cutiile sale.

Se mai poate întâmpla ceva neplăcut. Chiar dacă literele există, programul cu care lucrăm să nu interpreteze corect *input*-urile primite. Din păcate, acesta este cazul în mediul integrat T_EXnicCenter când este vorba despre fereastra cu mesaje de eroare. Literele românești nu sunt afișate corect.⁷⁵

B.3.1.1 Soluții L^AT_EX pentru limba română

L^AT_EX poate pregăti pentru tipar în condiții foarte bune texte cu litere românești. Voi prezenta mai întâi soluția pe care o folosesc efectiv în această carte și care mi se pare deosebit de comodă. Apoi voi descrie soluția clasică, tipică pentru spiritul T_EX, soluție care poate fi folosită pentru a pregăti surse L^AT_EX pe computere care nu au posibilitatea de a instala tastaturi cu litere românești.

B.3.1.1.1 Pachetele L^AT_EX și limba română Soluția folosită la pregătirea pentru tipar a cărții de față poate fi lesne înțeleasă examinând cele trei-patru rânduri din fișierul `sty` care-i spun sistemului L^AT_EX să folosească limba română:

```
1 \RequirePackage{type1ec}
2 \RequirePackage[T1]{fontenc}
3 \RequirePackage[cp1250]{inputenc}
4 \RequirePackage[english,romanian]{babel}
5 %=====
6 \newcommand{\texteng}[1]{\foreignlanguage{english}{#1}}
```

Primul rând cere sistemului să încarce pachetul `cm-super`. Acest pachet de peste 68MB conține o versiune a complet scalabilă a tipurilor de litere computer modern. Computer modern sunt literele clasice ale T_EX, proiectate de către însuși Donald Knuth. Ele nu sunt disponibile decât pentru un număr limitat de dimensiuni și rezistă mai prost la mărire. Pachetul `cm-super` este disponibil grație lui Vladimir Volovich.

⁷⁵N-am reușit să localizez problema, dar bănuiesc că totul pleacă de la problemele generate de ideea de integrare ca atare. Programul trebuie să captureze fluxul mesajelor compilatorului, care este apelat însă în linia de comandă.

Nu este absolut necesar să încărcăm cm-super. Puteți folosi literele standard în \LaTeX sau puteți încărca pachetul `times` pentru tipul de litere Times New Roman.

Al doilea rând este însă necesar pentru a avea acces la literele românești.⁷⁶ Comanda pe care o conține ar putea fi asemuită cu indicarea pe vremuri a cutiilor cu litere de plumb. Necesară pentru limba română este doar opțiunea `T1`.

Rândul al treilea conține o comandă care-i spune sistemului în ce „limbă” îi dăm ordinele privitoare la litere. În esență, opțiunea `cp1250` îi spune sistemului că noi folosim codificarea literelor din Windows.

Rândul al patrulea cere \LaTeX să încarce pachetul `babel`. Acest pachet, creat de Johannes L. Braams, permite \LaTeX să despartă corect în silabe în limba română, să scrie o serie de cuvinte în limba română. Căutați fișierul `romanian.ldf`; citiți și modificați (după ce v-ați făcut o copie de siguranță) unele elemente din acest fișier: eu unul prefer termenul „indice” (pentru englezescul *index*); „tabelul” și nu „tabela” și „glosar” pentru englezescul *glossary*.⁷⁷

Dacă nu funcționează ceva cum trebuie, citiți documentația. Nu faceți însă mai multe schimbări deodată. Faceți o singură schimbare și studiați efectele, pentru a detecta cauza fenomenului care vă deranjează.

Ultimul rând de cod listat mai sus din fișierul `sty` nu face decât să definească o comandă pentru delimitarea porțiunii de text care este în engleză. Atunci când încărcăm pachetul `babel` ultima opțiune este cea care se aplică automat.⁷⁸ Dacă vrem să folosim, într-o porțiune de text (un citat, de pildă), o altă limbă, trebuie să anunțăm în mod explicit acest lucru. Altfel, s-ar putea să avem diverse surprize neplăcute (la despărțirea în silabe, de exemplu).

B.3.1.1.2 Comenzi \LaTeX pentru diacritice \LaTeX are propriul său mod de a construi litere cu diacritice.⁷⁹ Există o serie de comenzi care ne permit să plasăm diacritice pe litere. Ca de obicei

⁷⁶ Programul de vizualizare a fișierului `dvi` n-ar afișa literele românești. Astfel se rupe ciclul editare-compilare-vizualizare.

⁷⁷ Nu uitați să spuneți sistemului MikTeX că folosiți limba română. A se vedea indicațiile de instalare a MikTeX la pagina 189, rândul 19.

⁷⁸ Același lucru este valabil și-n cazul pachetului `fontenc`!

⁷⁹ Pentru a folosi aceste comenzi n-aveți nevoie de pachete speciale. `Babel` continuă totuși să fie absolut recomandabil: fără acest pachet, ar trebui să indicați manual despărțirile corecte în silabe.

este mai simplu să recurgem la exemple. În cazul limbii române comenzile sunt:

```
\u{a} \u{A} \\
\^{} \^{} \\
\^{} \^{} \\
\c{s} \c{S} \\
\c{t} \c{T}
```

ă	Ă
â	Â
î	Î
ș	Ș
ț	Ț

Ca și-n cazul altor comenzi din \LaTeX , numele comenzii, chiar dacă se rezumă aici doar la un semn, este sugestiv.

5

B.3.2 Vim și adaptarea tastaturii

Dacă vrei să folosești comenzile specifice \LaTeX , crearea unui meniu Vim pentru literele românești cu diacritice nu este o soluție foarte practică. Există, din fericire, o alternativă. Schimbarea rolului unor taste.

10

Dacă fișierul `KBDRO.KBD` este criptic, fișierul de tip `vim` care face același lucru este relativ ușor de construit. În orice caz, soluția pe care o prezentăm în continuare este lesne de realizat de către oricine:

```
1 :imap <F2> \u{a}
2 :imap <S-F2> \u{A}
3 :imap <F3> \^{}
4 :imap <S-F3> \^{}
5 :imap <F4> \^{}
6 :imap <S-F4> \^{}
7 :imap <F7> \c{s}
8 :imap <S-F7> \c{S}
9 :imap <F8> \c{t}
10 :imap <S-F8> \c{T}
```

Să zicem c-ați scris un fișier `taste.vim` cu conținutul de mai sus. Experimentați o vreme într-un dosar special creat pentru teste. Creați un fișier. În modul normal dați comanda `:so taste.vim`.⁸⁰ Acum tastele funcționale au un rol!

15

Nu este nevoie de prea multe explicații pentru ceea ce ați obținut. În scriptul vim, cele două puncte sunt desigur un semn că este vorba despre o comandă care poate fi dată și-n linie. Puteți să le și eliminați. Cheia comenzilor este `imap`, care-i spune lui Vim să schimbe rolul unei taste în modul insert. `<F2>` se referă la cea de-a doua tastă funcțională.⁸¹ După numele tastei, este indicat noul ei rol.

20

⁸⁰ Alternativ, puteți folosi `Run a Vim Script` de pe bara cu instrumente.

⁸¹ Nu schimbați rolul tastei `F1`; ea servește la apelarea `help`-ului din Vim.

Eu recomand schimbarea în felul arătat mai sus a rolului tastelor funcționale. La urma urmei, pentru asta sunt disponibile pe tastatură: pentru a putea reprograma rolul lor.

Vă sâcăie însă inversarea rolului tastelor Y și Z din tastatura
 5 românească a Windows? n-ați găsit pe nimeni care știe cum să modifice KBDRO.RO? În Vim, puteți schimba rolul celor două taste. Studiați însă atentă sau atent următoarele două rânduri!

```
1 inoremap z y
2 inoremap y z
```

Dacă folosiți `imap` simplu, vă învârtiți în cerc! Vim va semnala o eroare. Comanda `inoremap` îi spune să nu *reschimbe* rolul tastei.

10 Dacă vă convine ideea, puteți continua experimentul cu modificarea rolului combinației SHIFT+Y, respectiv SHIFT+Z.

```
1 inoremap <S-z> Y
2 inoremap <S-y> Z
```

Dacă vreți să permanentizați soluții de genul celei de mai sus, nu uitați să plasați `taste.vim` în dosarul `plugin` din `vimfiles`. Verificați dacă nu se produc eventuale conflicte cu alte scripturi.

15 B.3.2.1 De la codurile Windows la comenzile \LaTeX și înapoi

Dacă studiați codurile unui fișier cu litere românești cu diacritice, veți vedea că literele minuscule cu diacritice au codurile hex următoare: `e3 e2 ee ba fe`. Traduse în baza 10, aceste coduri sunt: 227 226
 20 238 186 254. Cum folosește Windows aceste coduri? Gândiți-vă că este asemenea cuiva care ar avea un text criptat în față și ar căuta într-o carte în care pe fiecare pagină sunt rubrici de la 0 la 255; la fiecare rubrică găsește o dezlegare a codului care-l interesează.

B.3.2.1.1 Intermezzo pentru ucenicii vrăjitori Ați văzut
 25 probabil tot felul de programe care vă ajută să explorați casetele cu litere. Puteți descifra secretele care stau în spatele casetelor cu litere din Windows și folosind Vim.

Folosind posibilitatea de a trece de la vizualizarea textului la cea a codurilor hex nu vă va fi greu să construiți cu mâna un fișier care
 30 arată astfel în hex (în părțile sale semnificative):

1	00:	0102	0304	0506	0708	090d	0a0b	0c0d	0e0f
2	10:	1011	1213	1415	1617	1819	1a1b	1c1d	1e1f
3	20:	2021	2223	2425	2627	2829	2a2b	2c2d	2e2f
4	30:	3031	3233	3435	3637	3839	3a3b	3c3d	3e3f
5	40:	4041	4243	4445	4647	4849	4a4b	4c4d	4e4f
6	50:	5051	5253	5455	5657	5859	5a5b	5c5d	5e5f
7	60:	6061	6263	6465	6667	6869	6a6b	6c6d	6e6f
8	70:	7071	7273	7475	7677	7879	7a7b	7c7d	7e7f
9	80:	8081	8283	8485	8687	8889	8a8b	8c8d	8e8f
10	90:	9091	9293	9495	9697	9899	9a9b	9c9d	9e9f
11	a0:	a0a1	a2a3	a4a5	a6a7	a8a9	aaab	acad	aeaf
12	b0:	b0b1	b2b3	b4b5	b6b7	b8b9	babb	bcbd	bebf
13	c0:	c0c1	c2c3	c4c5	c6c7	c8c9	cacb	cccd	cecf
14	d0:	d0d1	d2d3	d4d5	d6d7	d8d9	dadb	dcdd	dedf
15	e0:	e0e1	e2e3	e4e5	e6e7	e8e9	eaeb	eced	eeef
16	f0:	f0f1	f2f3	f4f5	f6f7	f8f9	fafb	fcfd	feff
17	00:	0000	0d0a						

Reveniți apoi la vizualizarea în mod text. Aranjați simbolurile pe care le folosiți în texte în așa fel încât să fie ușor de înțeles codul hex din spatele lor. Eliminați din fișier primele 32 de coduri (dacă vreți să cercetați simbolurile ca atare).

În mediul verbatim semnele din sursă pot fi reproduse ca atare. Iată semnele de bază:

```

\begin{verbatim}
0 1 2 3 4 5 6 7 8 9 a b c d e f
2 0 ! " # $ % & ' ( ) * + , - . /
3 1 2 3 4 5 6 7 8 9 : ; < = > ?
4 @ A B C D E F G H I J K L M N O
5 P Q R S T U V W X Y Z [ \ ] ^ _
6 ' a b c d e f g h i j k l m n o
7 p q r s t u v w x y z { | } ~
\end{verbatim}

```

Restul semnelor nu pot fi
reproduse în mediul verbatim.

Vedeți în imaginea alăturată ce se întâmplă pe ecranul Vim.

```
e). 0123456789abcdef  
2 !"#$%&'()*+,-./  
3 0123456789:;<=>  
4 ABCDEFGHIJKLMNOP  
5 PQRSTUVWXYZ[\]^_  
6 `abcdefghijklmnopqrstuvwxyz  
7 ~  
8 € , f , , , t t % Š < Š Ž Ž  
9 ‘ ’ “ ” • — — ° ™ š > š ž ž  
a ~ Ł ł Ą Ć Œ © § « ¬ - ® Ž  
b ° ± } ´ µ ¶ • ¸ » L ˇ l ž  
c ř á â ã ä å Ľ Ć Č ě Ě ě Ě Ĩ Đ  
d Đ N Ń Ō Ő Ŏ Ů x Ř ů Ú Ů Ů Y T B  
e ř á â ã ä å Ľ Ć Č ě Ě ě Ě Ĩ Đ  
f đ ŋ ŋ ő ő ő ő ÷ ř ů ú ů ů y t .
```

Dacă veți combina cifra de pe prima coloană a rândului pe care se află litera cu cifra de primul rând al coloanei pe care se află simbolul care vă interesează, obțineți codul hex corespunzător.⁸²

Există însă o metodă mai simplă de a obține codul literei. Treceți editorul Vim în modul normal. Dați apoi comanda **ga**; pe ultima linie

⁸² Atenție! În hex, adică în baza 16, *a, b, c, d, e, f* sunt cifre, care se traduc în baza 10 prin *10, 11, 12, 13, 14*, respectiv prin *15*. Ceea ce vedeți în figură pe prima coloană și pe primul rând sunt cifre, nu litere!

a ecranului Vim vor apărea numele literei, codul zecimal al literei, codul în hex și-n octal.

Cu fișierul astfel creat puteți face investigații. Modificați versiunea tipului de literă și veți vedea alte semne pe ecran. Puneți, de pildă, versiunea pentru greacă a tipului de literă și observați unde apar literele grecești. Schimbați tipul de literă și studiați posibilitățile pe care vi le oferă.

B.3.2.1.2 Schimbări de care s-ar putea să aibă nevoie și persoanele obișnuite

Dacă sunteți o utilizatoare sau un utilizator obișnuit, v-ați mulțumit doar să citiți în diagonală secțiunea B.3.2.1.1. Dacă ați sărit peste ea, aruncați totuși o privire. Vă veți întreba poate de ce am folosit o imagine și nu un text pentru a ilustra felul în care Vim afișează literele limbilor Europei Centrale. Motivul rezidă în modul în care \LaTeX prelucrează *input*-ul pe care l-am produs cu ajutorul Vim.

Semnele de bază de care se folosește \TeX sunt între codurile 32 și 126, după cum se vede din tabelul de la pagina 246, rândul 5. Codul hex 20 corespunde codului zecimal 32. Este codul pentru un spațiu alb. Codul 127 generează $\langle \sim \rangle$.

Pachetul \LaTeX *inputenc* traduce, ca să spun așa, limbajul Windows în cel al \TeX . Aruncați o privire în fișierul *cp1250.def* și veți vedea definițiile folosite în procesul de traducere. Observați modul în care toate comenzile \TeX sunt scrise numai cu semnele de bază.

Ca și-n Vim, drept nume pentru simboluri voi folosi simbolul pus în paranteze unghiulare. Cu alte cuvinte, $\langle \sim \rangle$ este același lucru cu „tildă”. Codurile 127, 128 și 129 nu au o definiție în *cp1250*. Dacă vreți semnul pentru euro, folosiți comanda $\backslash\text{EUR}\{\}$ din pachetul *marvosym*. Există și variante ale acestei comenzi. Codul 130 este tradus prin $\backslash\text{quotesinglbase}\{\}$ și generează \langle , \rangle .

Codul 131 n-are o definiție în *cp1250*. Codul 132 este tradus prin $\backslash\text{quotedblbase}\{\}$ și generează $\langle ,, \rangle$.

Dăm în continuare restul listei codurilor care nu ridică probleme deosebite:

133 $\langle \dots \rangle$	134 $\langle \dagger \rangle$	135 $\langle \ddagger \rangle$	137 $\langle \% \rangle$	138 $\langle \S \rangle$
139 $\langle \langle \rangle$	140 $\langle \S \rangle$	141 $\langle \T \rangle$	142 $\langle \Z \rangle$	143 $\langle \Z \rangle$
145 $\langle \prime \rangle$	146 $\langle \prime \rangle$	147 $\langle \text{“} \rangle$	148 $\langle \text{”} \rangle$	149 $\langle \bullet \rangle$
150 $\langle \text{--} \rangle$	151 $\langle \text{—} \rangle$	153 $\langle \text{TM} \rangle$	154 $\langle \S \rangle$	155 $\langle \rangle$
156 $\langle \S \rangle$	157 $\langle \text{t} \rangle$	158 $\langle \Z \rangle$	159 $\langle \acute{z} \rangle$	160 $\langle \rangle$
161 $\langle \text{~} \rangle$	162 $\langle \text{~} \rangle$	163 $\langle \text{L} \rangle$	165 $\langle \text{A} \rangle$	167 $\langle \S \rangle$
168 $\langle \text{”} \rangle$	169 $\langle \text{©} \rangle$	170 $\langle \S \rangle$	171 $\langle \text{«} \rangle$	174 $\langle \text{®} \rangle$

175<Ž>	176<°>	178<„>	179<ł>	180<˘>
182<¶>	183<·>	184<„>	185<ą>	186<ş>
187<»>	188<Ł>	189<˜>	190<ŀ>	191<ž>
192<Ř>	193<Á>	194<Â>	195<Ă>	196<Ä>
197<Ĺ>	198<Č>	199<Ç>	200<Č>	201<É>
202<Ě>	203<Ě>	204<Ě>	205<Í>	206<Î>
207<Ď>	208<Đ>	209<Ñ>	210<Ñ>	211<Ó>
212<Ô>	213<Õ>	214<Ö>	216<Ř>	217<Ů>
218<Ú>	219<Ů>	220<Ü>	221<Ý>	222<Ť>
223<ſ>	224<ř>	225<á>	226<â>	227<ă>
228<ă>	229<Í>	230<ć>	231<ç>	232<č>
233<é>	234<ę>	235<ë>	236<ě>	237<í>
238<î>	239<ď>	240<d>	241<ń>	242<ň>
243<ó>	244<ô>	245<ő>	246<ö>	248<ř>
249<û>	250<ú>	251<ű>	252<ü>	253<ý>
254<ț>	255<˘>			

Codul 160 este tradus de `cp1250` prin `\nobreakspace{}`. Este normal ca-n tabel să apară doar un simplu spațiu alb. Uzual, în sursa `LaTeX`, punem mai degrabă o tildă pentru a indica spațiul insecabil.

Codul 173 este tradus prin `\-{}` care este comanda prin care i se spune sistemului `LaTeX` unde se *poate* face o despărțire în silabe. Nu l-am inclus în tabel pentru că efectul său este invizibil dacă nu se face efectiv o despărțire în silabe în punctul în care apare. De altfel, nici nu mi se pare o idee bună să folosim acest semn în sursa `LaTeX`; ar fi imposibil să-l deosebim de liniuța de unire. 5

Codul 136 are efect în Vim, dar este nedefinit în `cp1250`. În schimb, codurile 127 și 144 nu au efect în Vim și sunt nedefinite și-n `cp1250`. Folosirea lor generează o eroare. 10

Codurile următoare au efect în `LaTeX`, dar trebuie folosite în modul matematic, nu în mod text: 15

172<¬> 177<±> 181<μ> 215<×> 247<÷>

Cazurile cele mai interesante sunt cele ale codurilor 164 și 166. Ele sunt definite în `cp1250`. `LaTeX` nu generează un mesaj de eroare din perspectiva tastaturii, a *input*-ului. Apare însă un mesaj de eroare care ne spune că nu există în T1 simbolurile corespunzătoare. Este ca și cum turnătorii de litere de plumb ne-ar spune că n-are mulajele necesare pentru a turna litere de forma cerută. 20

Nu vă bucurați prea mult de listele cu coduri de mai sus. Ele sunt bune în cazul Windows. În lumea largă, veți întâlni o junglă de

astfel de coduri. Cu adevărat importantă este înțelegerea procesului în mai multe trepte prin care o apăsare pe o tastă se transformă într-un cod numeric, apoi într-o comandă `LATEX` și se termină prin găsirea unui simbol care este pus pe foaia de hârtie.

- 5 N-am putea opera și noi asemenea transformări? Desigur că da. În interiorul `LATEX` ele au o utilitate mai restrânsă, în măsura în care putem folosi pachete gata pregătite. Transformările sunt cu adevărat utile în afara `LATEX`.

- 10 Putem folosi Vim pentru a opera o trecere de la codificarea Windows la cea a `LATEX`. Voi ilustra acest lucru în cazul literelor românești. Generalizarea nu este dificil de realizat. Trebuie doar create niște fișiere de tip vim care conțin scripturile pentru realizarea substituițiilor. Iată cum facem trecerea de la coduri Windows la comenzi `LATEX`:

```
1 %s/ă/\u{a}/ge | update
2 %s/Ă/\u{A}/ge | update
3 %s/â/\~{a}/ge | update
4 %s/Â/\~{A}/ge | update
5 %s/î/\~{i}/ge | update
6 %s/Î/\~{I}/ge | update
7 %s/ș/\c{s}/ge | update
8 %s/Ș/\c{S}/ge | update
9 %s/ț/\c{t}/ge | update
10 %s/Ț/\c{T}/ge | update
```

- 15 Trecerea inversă este realizată în felul următor:

```
1 %s/\u{a}/ă/ge | update
2 %s/\u{A}/Ă/ge | update
3 %s/\~{a}/â/ge | update
4 %s/\~{A}/Â/ge | update
5 %s/\~{i}/î/ge | update
6 %s/\~{I}/Î/ge | update
7 %s/\c{s}/ș/ge | update
8 %s/\c{S}/Ș/ge | update
9 %s/\c{t}/ț/ge | update
10 %s/\c{T}/Ț/ge | update
```

- Acestea sunt scripturi Vim care trebuie chemate cu comanda `:so` urmată de numele scriptului în fișierul pe care vrem să-l transformăm. Alternativ, putem folosi **Run Vim Script** de pe bara cu instrumente. Este inutil, de asemenea, să adăugăm că trebuie să „vă faceți mâna” pe fișiere de probă.
- 20

B.3.2.1.2.1 Vrajitorii cu tastatura Tasta Q este, cel puțin din punctul meu de vedere, teribil de bine plasată pe tastatură și mi se întâmplă, în română, să o folosesc extrem de rar. În schimb, trebuie uneori să tastez de multe ori &, care mi se pare incomod plasat. Ca să rezolv situația dau, în modul normal al Vim, comanda `:imap q &` și tasta Q își schimbă ca prin minune rolul. 5

Cum scap însă de schimbarea de mai sus, fără să închid fișierul? Dau comanda `:iunmap q` și tasta Q revine la vechiul rol.

Puteți proiecta un script pentru a vrăji tastatura. Exemplul dat aici încearcă să arate cum pot fi corijate unele neajunsuri din tastatura românească standard în Windows. Schimbările sunt operabile numai în Vim! 10

Ideea de bază este cea de a atașa o altă codificare tastei. Următoarele comenzi vă permit să tasteți pe Ț în loc de { și pe ă în loc de [. 15

```
1 :map! { <char-195>
2 :map! [ <char-227>
```

Puteți scrie și ă în loc de <char-227> sau puteți folosi notația în hex <char-0xe3>. Cifrele în hex trebuie precedate de 0x.

Dacă știm codurile pe care vrem să le introducem de la tastatură, nu este greu să schimbăm rolul tastelor.

Putem modifica și rolul unei combinații de taste: 20

```
1 :map! /a ă
```

Trebuie să bateți însă suficient de repede tastele una după alta pentru a obține efectul dorit.

Putem crea și scripturi cu astfel de modificări ale tastelor. Atenție însă la asigurarea unui gen sau altul de mecanism de scăpare. Puneți, de exemplu, în script ceva de genul: 25

```
1 :map! <F5> <Esc>:unmap! [<CR>
```

tastatură Vim are posibilitatea de a crea tastaturi cu ajutorul unor scrip-
româ-turi speciale. Doar cu titlu de exemplu, putem presupune c-am creat
nească un fișier `romana.vim` și l-am plasat în dosarul `keymap` din `vimfiles`.
pentru Conținutul său este următorul:
Vim

```
1 let encoding='cp1250'
2 let b:keymap_name = "ro"
```

```

3 loadkeymap
4 " Taste modificate în jurul lui Enter:
5 [ ă
6 { Ă
7 ] î
8 } Î
9 \\ â
10 <Bar> Â
11 ; §
12 : Ș
13 ' ț
14 \" Ț
15 " Recuperarea tastelor pierdute (folosind combinații):
16 /. :
17 /, ;
18 /q {
19 /w }
20 /z \\

```

Încărcăți tastatura cu comanda `:set keymap=romana` sau folosind Keymap din meniul Edit. Puteți comuta între tastatura românească și cea standard cu `CTRL+^`.

- Tastatura astfel definită este mult mai flexibilă decât cea din Windows. Poziția literelor pe taste poate fi modificată ușor. Semnele sacrificate prin plasarea de litere românești pot fi regăsite prin combinații de taste. Aceste combinații pot fi adaptate la gusturile fiecărei persoane care utilizează Vim.

B.3.3 Limbile europene care folosesc alfabetul latin

- 10 Să vedem acum câteva exemple de nume, titluri și expresii care pot fi întâlnite într-un text filosofic. Începem nume, titluri sau expresii de sorginte franceză sau germană:

Br\'{e}hier\\	Bréhier
L'\^{E}tre et le n\'{e}ant\\	L'Être et le néant
La science et l'hypoth\'{e}se\\	La science et l'hypothèse
G\'{o}del\\	Gödel
f\'{u}r sich	für sich

- 15 Italiana, spaniola sau portugheza pun și ele probleme dacă vrem să reproducem corect un titlu, un nume sau o expresie:

0,1 e 2 e cos\'{i} via\\	0,1 e 2 e così via
Espa\'{n}a inteligible\\	España inteligible
Inicia\c{c}\^{a}o ao filosofar	Iniciação ao filosofar

S-ar putea ca mai rar și mai puțină lume să se ciocnească de problemele ridicate de limbile slave care folosesc alfabetul latin. Li-

teratura poloneză de logică și de filosofie este totuși foarte cunoscută și voi alege de acolo câteva exemple:

```
semiotyka j\k{e}zyk\'{o}w
naturalnych\\
metaj\k{e}zyk\\
Tadeusz Cze\{z}owski\\
Izydora D\k{a}mbska\\
mo\{z}liwo\'{s}\\'{c}
```

```
semiotyka języków naturalnych
metajęzyk
Tadeusz Czeżowski
Izydora Dąmbska
możliwość
```

Maghiara ridică și ea probleme specifice atunci. Fonemele ö și ü au și variante lungi:

```
az igazh\'{i}v\H{o}k\\
m\H{u}velet
```

```
az igazhívók
művelet
```

Unele limbi europene au simboluri speciale, pe lângă cele din alfabetul latin:

```
\L{ }ukasiewicz\\
Bew\ss{ }tsein\\
S{o{ }ren Kierkegaard\\
\AA{ }kvist\\
filozof ta\c{s}\i{ }\\
\textquestiondown{ }
humanizaci\'{o}n
o deshumanizaci\'{o}n?\\
\textexclamdown{ }\ldots{ }!
```

```
Łukasiewicz
Bewußtsein
Søren Kierkegaard
Åkvist
filozof taşı
¿humanización o deshumaniza-
ción?
¡...!
```

Spaniola împrăntează întrebările și propozițiile exclamative. Dacă ne gândim bine, procedura este cât se poate de rațională; atâta doar că avem nevoie de două simboluri speciale.

Comenzile \LaTeX îngropate, ca să spun așa, în cuvinte sunt notate în versiunea pe care o prefer eu. Cred că sursa este mai lizibilă dacă punem sistematic argumentele unei comenzi în acolade și, în cazul în care nu există un argument, punem doar acolade.⁸³ Consultați documentația pentru a vedea și alte posibilități de a nota comenzile respective.⁸⁴

Modul acesta de a cere prin comenzi speciale simboluri este util în special atunci când scriem nume străine, expresii din alte limbi sau texte foarte scurte. n-are rost atunci să punem în mișcare toată mașinăria descrisă în §B.3.1.1.1 pentru a pune un accent pe o literă.

⁸³ Atenție la rolul acoladelor chiar și atunci când nu conțin nimic! Fără ele, \LaTeX nu știe unde se termină comanda. Alternativa este un spațiu alb, dar acest spațiu mi se pare derutant pentru ochiul uman. Am mai putea pune în acolade întreaga comandă.

⁸⁴ Vedeți în special Scott Pakin *The Comprehensive \LaTeX Symbol List* în dosarul `texmf/doc/latex/comprehensive` din documentația MikTeX.

B.3.4 Tehnica alegerii tipului de literă

Până acum am pus un mare accent pe modul în care *vedem* ceea ce introducem de la tastatură. Am descoperit că literele scrise pe tastatura fizică sunt de fapt niște nume pentru tastele fizice. Ceea ce

5 introducem în realitate de la tastatură sunt niște coduri numerice. Putem vizualiza în mod comod aceste coduri pe ecranul computerului.

Cum ideea fundamentală aici este aceea de a separa diversele procese pentru a le înțelege și dirija mai bine, n-are rost să afișăm

10 pe ecran rezultatul final (*output*)-ul, ca-n sistemul WYSIWYG. Să vedem însă cum putem controla aspectul final al literelor.

B.3.4.1 Modificarea aspectului *output*-ului

Pentru unii, cărțile în limba germană scrise cu „caractere gotice” sunt ceva misterios, texte pe care nimeni nu pare a le putea citi.

15 Pentru alții, este vorba de un adevărat obiect de cult. \LaTeX poate crea tipărituri pentru multe gusturi, după cum se vede din exemplele următoare:

```
{%start grup local
\fontfamily{yfrak}
\fontsize{19}{22}\selectfont
Eseul filosofic
\par}%încheiere de grup cu \par
```

Eseul filosofic

```
{
\fontfamily{phv}\fontseries{b}
\fontsize{14}{17}
\selectfont\scshape
Eseul filosofic
\par}
```

ESEUL FILOSOFIC

```
{
\fontfamily{ptm}
\fontsize{17}{21}\selectfont
Eseul filosofic
\par}
```

Eseul filosofic

```
{
\fontfamily{pzc}
\fontsize{17}{20}\selectfont
Eseul filosofic
\par}
```

Eseul filosofic

```
{
\fontfamily{pcr}
\fontsize{16}{19}\selectfont
Eseul filosofic
\par}
```

Eseul filosofic

```
{
\fontfamily{cmss}
\fontsize{17}{22}\selectfont
Eseul filosofic
\par}
{
\fontfamily{cmr}
\fontsize{17}{21}\selectfont
Eseul filosofic
\par}
```

Eseul filosofic

Eseul filosofic

Ar fi multe comentarii de făcut în legătură cu tipurile de litere exemplificate mai sus. Numele lor în T_EX reprezintă o prescurtare. Ultimul exemplu este cel al **cmr**, tipul de literă creat de către Donald Knuth și destinat în mod special sistemului T_EX. Numele propriu-zis al acestui tip de literă este „computer modern“. Varianta sa fără serife este ilustrată de penultimul exemplu. 5

Puteți compara tipul de litere **cmr** cu clasicul tip *times*, ilustrat de către cel de al treilea exemplu. Consultați documentația MikT_EX pentru a afla mai multe despre diversele tipuri de litere. 10

Comanda `\fontsize{}{}` are două argumente. Primul argument este cel care indică dimensiunea literei. Al doilea reglează poziția literei în cutia T_EX.

B.3.4.2 Alfabetul chirilic 15

Experimentul este cea mai bună metodă de a învăța. Cu condiția să nu lucrați cu fișiere pe care n-ați vrea să le distrugeți. Schimbați opțiunea deci opțiunea pachetului **fontenc** în OT2. Să vedem ce s-ar întâmpla în acest caz cu semnele folosite în mod clasic de către T_EX (vezi pagina pagina 246, rândul 5): 20

```
! " % ' ' ( ) * , - . /
0 1 2 3 4 5 6 7 8 9 : ; < > ?
А Б Ц Д Е Ф Г Х И Ј К Л М Н О
П Ч Р С Т У В Щ Ш Ъ З [ ]
‘ а б ц д е ф г х и ј к л м н о
п ч р с т у в щ ш ѝ з | 25
```

S-ar părea că a avut loc un accident. Aveți în sursă aceleași coduri, dar rezultatul final apare cu litere chirilice. Nu a avut loc nimic neobișnuit. Codificarea finală este cea care s-a schimbat.

Dacă în fișierul **sty** facem câteva modificări putem insera în textul cu litere latinești text cu litere chirile: 30

```
\RequirePackage[OT2,T1]{fontenc}
```

```

2 %.....
3 \newcommand{\textrus}[1]{\foreignlanguage{russian}{#1}}
4 \newcommand{\dur}{\cyrhrdsn}
5 \newcommand{\Dur}{\CYRHRDSN}
6 \newcommand{\moale}{\cyrstfn}
7 \newcommand{\Moale}{\CYRSFTSN}
8 \newcommand{\erus}{\cyrrrev}
9 \newcommand{\Erus}{\CYRREV}

```

Sistemul este util, de exemplu, pentru a scrie titluri cu chirilice. Iată codul pentru titlul cărții lui Dubovik, Veize și Golovko:

```

\textrus{Referirovanie i
annotirovanie
{\erus}konomicheskikh tekstov
na angli\u{i}skom yazyke}

```

Реферирование и аннотирование экономических текстов на английском языке

Soluția aceasta este ideală atunci când textul nu este preponderent în limba rusă sau nu dispunem de cunoștințele necesare pentru a modifica tastatura. Secretul ei este că o comandă precum:

```

1 \textrus{
2 abvgde\"{e}zhzi\u{i}klmnoprstufkhkhtschshshch
3 \dur{}y\moale{}\erus{}yuya}

```

are drept rezultat la ieșire, pe foaia de hârtie (fizică sau electronică) șirul: абвгдеёжзийклмнопрстуфхцчшщъыьэюя de semne ale alfabetului rus.⁸⁵

Dacă majoritatea textului este cu chirilice pare mai rațional să modificați codificarea *input*-lui. În acest caz, puteți folosi opțiunea `cp1251` pentru *inputenc* și o opțiune de genul `T2A` pentru *fontenc*. Puteți avea în continuare acces la literele românești cu ajutorul comenzilor \TeX pure. De asemenea, trebuie să modificați tastatura Vim în așa fel încât să puteți introduce lesne literele chirilice.⁸⁶

B.3.5 Limba greacă veche

Ar fi inutil să explic aici importanța limbii grecești vechi pentru filosofie. Nu puține sunt eseurile filosofice în care apar termeni în greaca veche sau citate în greaca veche.

Pot spune din proprie experiență că transliterarea nu este o soluție fericită. În dicționarul termenilor filosofici grecești,⁸⁷ un cuvânt

⁸⁵Pentru detalii citiți documentația din dosarul `\latex\cyrillic` din documentația $\text{Mik}\TeX$.

⁸⁶Detalii și alte exemple vor fi afișate treptat pe situl cărții.

⁸⁷Francis E. Peters, *Termenii filozofiei grecești* (București: Humanitas, 1993), p.92.

precum ἐπιστήμη apare transliterat ca **epistēmē**. Spiritul textului grecesc vechi este pierdut la propriu și la figurat. Mai rău, εἶδος devine **eidos**, ceea ce este de-a dreptul greșit.⁸⁸

Există o transliterare precisă a textelor grecești vechi. Ea se numește „betacod“. Transliterarea aceasta n-are un aspect estetic prea plăcut, dar este foarte eficientă în rolul de *input* într-o sursă L^AT_EX.

Pentru a folosi betacodul trebuie să încărcăm pachetul **betacode** punând în locul cuvântului **babel** cuvântul **betababel**. Pachetul **betababel** este o extensie a pachetului **babel** creată de către Felix G. Berkemeier.

Voi explica principiile de bază ale betacodului cu ajutorul unor termeni filosofici foarte cunoscuți:⁸⁹

```
\bcode{A}DIA/FORON}\\
\bcode{AU)TA/RKEIA}\\
\bcode{AU)TARXEI/A}
```

ἀδιάφορον
αὐτάρχεια
αὐταρχεία

Betacodul nu deosebește între majuscule și minuscule. Există obiceiul de a folosi masiv majusculele englezești. Transliterarea literelor ca atare este evidentă. De altfel, vom da exemple pentru fiecare literă din alfabetul clasic grecesc.

Spiritul lin este transliterat prin paranteza rotundă închisă. Spiritul aspru este transliterat prin paranteza rotundă deschisă. Accentul ascuțit este transliterat printr-o bară oblică. Accentul grav este transliterat printr-o bară oblică inversă. Comanda `\bcode{}` inserează textul în punctul în care este dată.

Fiiți atente și atenți la diferența dintre ultimele două cuvinte de mai sus. Primul înseamnă *independentă absolută* (autosuficiență), iar al doilea *putere absolută*. Nu traduceți al doilea termen de mai sus printr-o transliterare aproximativă a celui de-al treilea termen.⁹⁰

```
\bcode{BOU/LHSIS}\\
\bcode{GE/NESIS}\\
\bcode{DIA/NOIA}\\
\bcode{EI)=DOS}
```

βούλησις
γένεσις
διάνοια
εἶδος

Pachetul **betababel** deosebește automat între sigma intern și fi-

⁸⁸În toată cartea se folosește doar un singur tip de accent, iar în glosarul român-elin de la sfârșit diacriticele grecești dispar cu totul! Pentru forma corectă a lui εἶδος și explicarea poziției spiritului a se vedea Ana Felicia Ștef, *Manual de greacă veche* (București: Humanitas, 1996), p.23.

⁸⁹Extrași tot din Peters, *op.cit.*, dar transliterați în betacod și prelucrați în L^AT_EX.

⁹⁰Din păcate, ediția 1993 din Peters, *op.cit.* conține exact această greșală (v.p.52).

nal. Dacă avem nevoie, putem însă scrie `\bcode{s1}` pentru a obține σ și `\bcode{s2}` pentru a obține ς .

`\bcode{ZW=|ON}`

ζῶον

Accentul circumflex este notat prin semnul egalității. Iota subscris este notat printr-o bară verticală.

`\bcode{H(DONH/)}\ \`
`\bcode{QEWR|/A}\ \`
`\bcode{I|SONOMI/A}\ \`
`\bcode{K|O/SMOS}\ \`
`\bcode{L|O/GOS}\ \`
`\bcode{TA\ MAQHMATIKA/}`

ἥδονή
 θεωρία
 ἰσονομία
 κόσμος
 λόγος
 τὰ μαθηματικά

`\bcode{NOU=S}\ \`
`\bcode{CEN|O/S}\ \`
`\bcode{O|)/NOMA}\ \`
`\bcode{PAQO/S}\ \`
`\bcode{R(OH/)}\ \`
`\bcode{SWFROSU/NH}\ \`
`\bcode{TE/LOS}\ \`
`\bcode{U(P|O/QESIS)\ \`
`\bcode{FRO/NHSIS}\ \`
`\bcode{XRO/NOS}\ \`
`\bcode{YUXH/}\ \`
`\bcode{W(/RA}`

νοῦς
 ξενός
 ὄνομα
 παθός
 ῥοή
 σωφροσύνη
 τέλος
 ὑπόθεσις
 φρόνησις
 χρόνος
 ψυχή
 ὥρα

Betacodul este, în general, transparent. Unele transliterări pot fi mai dificil de descifrat. Cred c-ar fi mai greu, la prima vedere, să vă dați seama, văzând numai betacodul, că penultimul cuvânt este familiarul termen grecesc pentru minte sau suflet.

În cazul literelor majuscule, regula betacodului este ca semnele diacritice să fie puse înaintea literei. Pachetul **betababel** este flexibil și admite punerea diacriticelor și după literă, cu condiția să fim consecvenți. Voi testa posibilitățile pachetului cu câteva nume proprii.⁹¹ Voi devia, de asemenea, de la practica folosirii exclusiv a majusculilor.

`\bcode{*Aristote/lhs}\ \`
`\bcode{*Dioge/nhs *Lae/rtios}\ \`
`\bcode{*E|mpedoklh=s}\ \`
`\bcode{*Pla/twn}`

Ἀριστοτέλης
 Διογένης Λαέρτιος
 Ἐμπεδοκλῆς
 Πλάτων

Pentru porțiunile ample de text, pachetul **betababel** oferă un

⁹¹Numele sunt extrase din manualul Anei Ștef, *op.cit.*

mediu L^AT_EX special. Voi ilustra utilizarea acestui mediu cu un citat din Aristotel.⁹²

συλλογισμὸς δὲ ἐστὶ λόγος ἐν ᾧ τεθέντων τινῶν ἕτερόν τι τῶν
κειμένων ἐξ ἀνάγκης συμβαίνει τῷ ταῦτα εἶναι.

Sursa în L^AT_EX a citatului este următoarea:

5

```
1 \begin{betacode}
2 sullogismo\s de/ e)sti lo/gos e)n w(=| teqe/ntwn tinw=n e(/tero/n
3 ti tw=n keime/nwn e)c a)na/gkhs sumbai/nei tw=| tau=ta ei)=nai.
4 \end{betacode}
```

B.3.5.1 Pachetul babel și limba greacă

Betacodul este un mod comod și precis de a introduce text în limba greacă veche. După cum spuneam, el nu este însă decât un etaj ridicat peste clădirea mai vastă a pachetului babel.

Vom coborî pentru o clipă sub etajul betababel-ului ca să vedem ce se petrece în cadrul babel.

10

Pentru început faceți experimente de genul următor:

```
\documentclass{article}
\usepackage[cp1250]{inputenc}
\usepackage[LGR]{fontenc}
\begin{document}
Swkráthc
\end{document}
```

Σωκράτης

Observați cum, ca și-n cazul literelor chirilice, am modificat codificarea la ieșire, nu pe cea de la intrare. Este normal să procedăm așa atâta timp cât suntem interesate și interesați de inserarea unor cuvinte sau mici fragmente de text în greacă într-un text care altfel este scris preponderent cu caractere latine.

15

Dac-ați sesizat că *a* cu accent ascuțit are codul 225, atunci sunteți pe drumul cel bun cu experimentele. Pasul următor constă în localizarea efectului obținut mai sus. Scrieți un alt proiect, în care preambulul arată astfel:

20

```
1 \documentclass{article}
2 \usepackage[cp1250]{inputenc}
3 \usepackage[LGR,T1]{fontenc}
4 \usepackage[polutonikogreek,romanian]{babel}
5 \newcommand{\textelin}[1]{\foreignlanguage{polutonikogreek}{#1}}
```

⁹²Aristotel, An.pr.A 1.24^b 18-20. Citat după textul grecesc editat de W.D.Ross, *Aristotelis analytica priora et posteriora*(Oxford: Oxford University Press, 1968).

Puneți numele lui Socrate între acoladele comenzii `\textelin{}`. Acum doar acest cuvânt va apărea cu litere grecești. Opțiunea LGR nu este acum neapărat necesară. De fapt, pachetul `babel` va face oricum apel la ea.

- 5 Acum ar trebui să fie și mai limpede ce rol joacă `babel` și ce rol joacă `betababel`. Când încercăm `betababel` acest pachet cheamă pachetul `babel` și introduce opțiunea `polutonikogreek`. De aceea, opțiunea aceasta este nu doar superfluă, ci generează chiar erori în cazul în care am încărcat pachetul `betababel`.

10 B.3.5.2 Ibycus

- O codificare similară cu `betacodul` este cea folosită de către pachetul `ibycus`.⁹³ Trebuie să instalați însă separat acest pachet. El nu este inclus în distribuția `MikTeX`. Este absolut recomandabil să-l puneți în dosarul `localtexmf`, pentru a nu-l confunda cu pachetele standard `MikTeX`. Folosiți separat pachetul `ibycus` separat de alte pachete, în documente destinate cu preponderență producerii de texte tipărite frumos în greaca veche.⁹⁴

Iată un fragment, tot din Aristotel,⁹⁵ dar în codificarea `ibycus`:

```
1 le'gw de' oi(=on a)na'gkh me'n e)'sesqai naumaxi'an au)'rion h)'
2 mh' e)'sesqai, ou) me'ntoi gene'sqai au)'rion naumaxi'an
3 a)nagkai=on ou)de' mh' gene'sqai; gene'sqai me'ntoi h)' mh'
4 gene'sqai a)nagkai=on.S
```

- Asemănările cu `betacodul` sunt limpezi. Există doar câteva deosebiri
20 majore. `Ibycus` folosește distincția dintre minuscule și majuscule (nu steluța pentru a marca majusculele). De asemenea, notarea accentelor ascuțit și grav este diferită.

- Folosirea semnului ; este, de asemenea, diferită. În `betacod`, după obiceiul grecesc, acesta este semnul întrebării. `Ibycus` îl folosește pentru punctul ridicat. Afișat cu litere grecești (folosind însă `betababel`) textul arată astfel:

⁹³Pachetul a fost creat de către Pierre A. MacKay (University of Washington) pe baza unui tip de literă produs de către Silvio Levy. Levy a încercat să mențină tipul său de literă cât mai aproape de tipul de literă Didot, care-și are originea la începutul secolului al XIX-lea.

⁹⁴`Ibycus` poate fi folosit împreună cu opțiunea `OT1` a pachetului `fontenc`, dar nu cu `T1`. Practic, `Ibycus` este de folosit când vreți să produceți text numai în greacă.

⁹⁵Este vorba de un fragment din celebrul pasaj despre bătălia navală care va avea loc mâine. Vezi Aristotel, *De Interpretatione* 19a 29-32.

λέγω δὲ οἷον ἀνάγκη μὲν ἔσεσθαι ναυμαχίαν αὐριον ἢ μὴ ἔσεσθαι, οὐ μέντοι γενέσθαι αὐριον ναυμαχίαν ἀναγκαῖον οὐδὲ μὴ γενέσθαι· γενέσθαι μέντοι ἢ μὴ γενέσθαι ἀναγκαῖον.

S-ar putea să vreți să folosiți `ibycus` din pricina unor probleme ale pachetului `betababel`.⁹⁶

5

B.3.6 \LaTeX și unicode

Dacă scrieți într-un fișier de tip text, a apăsa pe o tastă la computer nu este totuna cu a pune o ștampilă sau cu a imprima o literă cu mașina de scris. Ceea ce pune, de fapt, în fișier computerul este un cod numeric. Până acum am văzut doar coduri formate din două cifre hex. În total există 256 de astfel de coduri distincte.

10

Dacă-ți studiat cumva tabelul de la pagina 246, rândul 5, ai observat că nu toate cele 256 de coduri posibile sunt folosite pentru semnele diverselor alfabet. Motivul este foarte simplu: într-un fișier text este nevoie, de pildă, de coduri pentru sfârșitul de rând, precum și de alte coduri speciale. Chiar și așa rămân destule coduri pentru limbile europene.

15

Toate limbile Europei folosesc un alfabet de un tip sau altul. Nu este posibil să fie puse toate semnele lor în corespondență un o singură pagină cu coduri de genul celor descrise mai sus. Faptul că folosim mai multe pagini cu coduri reprezintă motivul pentru care trebuie să specificăm codificarea *input*-ului. Ce se întâmplă însă în cazul unei scrieri ideografice? Ar fi greu să tot schimbăm pagina cu coduri la fiecare pas.

20

Ideea unicodului (a unei codificări universale) este să folosim patru cifre hex în loc de două. În acest caz, codul ultim din tabel ar fi `ffff`, ceea ce corespunde în sistemul zecimal cu `65535`. După cum se vede, diferența este uriașă. Dar și jungla creată este imensă; totul trebuie parcurs pas cu pas.

25

B.3.6.1 Pachetul `ucs`

30

După cum se vede mai sus, în cazul scrierilor care folosesc un alfabet, ne putem descurca excelent separând porțiunile în care folosim o anumită codificare de celelalte. Cadrul oferit de codurile de la 0 la `FF` este totuși strâmt. Pentru a avea acces la lumea codificării unicode,

⁹⁶Sigma majusculă pune probleme în `betababel`. Urmăriți situl cărții pentru a detalii, exemple și eventuale soluții îmbunătățite.

trebuie recurs la pachetul ucs. Încărcarea sa se face după modelul următor:

```

1 \usepackage{ucs}
2 \usepackage[utf8,cp1250]{inputenc}
3 \usepackage[LGR,OT2,T1]{fontenc}
4 \usepackage[romanian]{betababel}
5 \newcommand{\textelin}[1]{\foreignlanguage{polutonikogreek}{#1}}

```

Foarte important este faptul că pachetul ucs este încărcat înaintea pachetului inputenc. Observați, de asemenea, opțiunea utf8 la pachetul inputenc. Această opțiune cere sistemului să preia și coduri unicod. Ordinea opțiunilor este foarte importantă. Ultima opțiune este cea care are statut de default (este executată automat).

Puteți folosi în continuare metodele descrise mai sus sau chiar le puteți combina precum în acest fragment din Platon:⁹⁷

```

1 \begin{quote}
2 \textelin{S}\bcode{WKRA/THS SOFO\S A)NH/R,
3 TA/ TE METE/WRA FRONTISTH\S KAI\}\ldots
4 \end{quote}

```

10 Σωκράτης σοφὸς ἀνὴρ, τὰ τε μετέωρα φροντιστὴς καὶ ...

Codificarea amplă care este unicodul e imposibil de descris pe larg aici.⁹⁸ Principiul după care este construit pachetul ucs este acela al folosirii comenzilor L^AT_EX disponibile în diverse pachete L^AT_EX.⁹⁹ Practic, un cod unicod este tradus printr-o comandă L^AT_EX care, la rândul ei, generează comenzile necesare pentru a produce semnele (evident, dacă sunt disponibile în L^AT_EX).

Voi folosi tot exemple legate de limba greacă veche pentru a ilustra felul în care se aplică principiul de mai sus.

Să spunem c-ați consultat tabelele unicod și ați văzut că litera alfa are codul hex 03b1. Faceți calculele¹⁰⁰ pentru a trece în sistemul zecimal și veți afla că alfa minusculă are codul 945. Aceste numere de cod pot fi folosite ca-n exemplul care urmează.

⁹⁷Platon, Apologia 18b 7.

⁹⁸Puteți consulta listele de coduri utilizabile cu pachetul ucs la adresa de Internet <http://www.unruh.de/DniQ/latex/unicode/tables/index.html>. Listele sunt alcătuite chiar de către Dominique Unruh, care a realizat pachetul ucs.

⁹⁹Atenție la încărcarea lor atunci când este necesar!

¹⁰⁰Puteți folosi calculatorul din accesoriile Windows sau un program special de convertire de unități de măsură și numere, cum este Versaverter de la <http://pawprint.net>, care este un program gratuit.

```
\textelin{\unichar{945}}\
\unichar{946}}\
\unichar{947}}
```

α

β

γ

Așa cum spuneam, sistemul unicond este, de fapt, transpus în comenzi \LaTeX . De unele dintre ele s-ar putea să aveți nevoie, mai ales în cazuri speciale. Iată două exemple:

```
\textelin{\textdigamma}}\
\textelin{\textsanpi}}
```

ƒ
λ

5

Ultimul simbol are codul hex 03e1 (zecimal 993). Alte coduri și comenzile aferente lor pot fi studiate cu ajutorul documentației pachetului ucs sau citind chiar programele din pachet ca atare.

B.3.7 Alte pachete cu simboluri în \LaTeX

Unicond are ample posibilități, dar cred că nu trebuie să exagerăm cu recursul la unicond. Dacă scriem într-o limbă europeană, s-ar putea să folosim una-două pagini cu coduri și n-are rost să încărcăm zeci de mii de simboluri pentru a folosi câteva zeci.




Persoanele pasionate de gândirea orientală, de culturile vechi vor fi desigur atrase de unicond și posibilitățile acestuia. Dar chiar și acestor persoane nu trebuie să le scape pachetele care s-ar putea să se potrivească mai bine cu obiectivele lor.

Favoritele mele sunt pachetele concepute de către Peter Wilson pentru unele scrieri arhaice.¹⁰¹ Pachetele acestea sunt simple, se compilează ușor și sunt bune pentru cineva care are cunoștințe extrem de limitate despre scrierile respective. Exemplul care urmează provine din scrierea vechilor egipteni.

Hieroglife



este un cuvânt care se transcrie fonetic *rh*. Înseamnă *a ști*.

Simbolul  este o gură stilizată. Simbolul  este o sită. Aceste simboluri sunt folosite aici cu valoarea lor fonetică, dar Christian Jacq arată că erau posibile asocieri cu transmiterea preponderent verbală a cunoașterii și cu procesul de selecție, de cernere a ideilor.¹⁰² Semnul  este un determinativ pentru conceptele abstrakte. Christian Jacq sugerează că prin inversarea sitei și a gurii obținem un interesant joc de cuvinte. Termenul rezultat, care se transcrie

¹⁰¹ A se vedea documentația creată pentru aceste pachete de către Peter Wilson <peter.r.wilson@boeing.com> în dosarul `\texmf\doc\doc\fonts\archaic`.

¹⁰² Christian Jacq *Sag's mit Hieroglyphen: Lesen und Schreiben wie die alten Ägypter* (Hamburg: Rowohlt, 2003), p.169.

fonetic *hr*, înseamnă *a cădea*. Determinativul este acum un om aflat în cădere, un om care, ignorant fiind, este condamnat la decădere.¹⁰³

Cu pachete \LaTeX puteți scrie și-n armeană, georgiană, sanscrită, chineză și multe alte limbi. Pachetele au sursele deschise și sunt chiar
5 mai ușor de folosit decât un program cu surse ascunse.

Nu găsiți literele sau simbolurile de care aveți nevoie? Donald Knuth a creat un limbaj special pentru generarea de tipuri de litere (*fonturi*). Limbajul se numește METAFONT.¹⁰⁴ Pentru hieroglifele
egiptene din pachetul lui Peter Wilson vedeți fișierul `pmhg.mf` din
10 dosarul `\texmf\fonts\source\public\archaic`. METAFONT

În principiu, dacă ați dobândit competența necesară, în \LaTeX puteți include orice simbol. Desigur, la nevoie, trebuie să creați simboluri. Este însă foarte probabil că, în lumea largă a utilizatorilor de \LaTeX , există deja cineva care a conceput o soluție la problema
15 care vă frământă.

B.4 Tabele și formule

Multe dintre eseurile filosofice nu au deloc tabele sau formule. Persoanele care folosesc intens analiza logică au însă nevoie cel puțin de cunoștințele de bază privitoare la scrierea de formule. De aseme-
20 nea, multe eseuri de filosofie politică recurg la tabele statistice.

Voi începe cu tabelele pentru că înțelegerea modului în care se construiesc și se folosesc tabelele va ajuta mult la formarea deprinderilor necesare pentru a scrie formule complexe.

B.4.1 Principiile de bază ale construirii tabelelor

25 Tipul cel mai simplu de tabel conține doar text aranjat pe linii și coloane. Voi descrie întâi modul de a construi un mic tabel, pe care-l voi folosi mai jos pentru a realiza un tabel ceva mai complicat.

Tabelul simplu are două coloane și două rânduri. Nu servește decât la dispunerea a patru cuvinte în cele patru colțuri ale unei
30 mici pagini.

```
\begin{tabular}{cc}
Liberal & Libertarian \\
Populist & Conservator \\
\end{tabular}
```

Liberal	Libertarian
Populist	Conservator

Tabelul este construit într-un mediu denumit `tabular`. După

¹⁰³ *Ibidem.*

¹⁰⁴ A se vedea aici C.3.

\begin{tabular} trebuie adăugată o pereche de acolade între care este precizată alinierea conținutului coloanelor. Alinierea este indicată cu ajutorul literelor l, c, r (pentru aliniere stânga, centru, respectiv dreapta). În cazul nostru, ambele coloane au textul centrat. Pe fiecare rând, coloanele sunt separate prin semnul &. Capetele de rând sunt indicate prin \\ și este important ca fiecare rând să aibă exact numărul de coloane specificat.

Pentru a crea tabele sau scheletul unor tabele ar fi bine însă, la început cel puțin, să folosiți un program de creare a tabelelor în L^AT_EX. Puteți folosi, de pildă, LaTeX. Acesta este un program gratuit, creat de către Alex A. Denisov¹⁰⁵. Interfața grafică a programului permite crearea tabelului fără a recurge direct la comenzile L^AT_EX.

Un tabel este în fapt rodul unui mic proiect. Ca la orice proiect, este necesar să precizați obiectivele. Căutați apoi mijloacele cele mai potrivite pentru a atinge obiectivele fixate. Desfaceți totul pe module și construiți fiecare modul separat, pentru a nu fi victime ale complexității tabelului.

Proiectul meu aici este să reproduc un tabel cu tipurile de ideologii, creat de către William S. Maddox și Stuart S. Lilie.¹⁰⁶ Miezul tabelului mai mare îl constituie micul tabel de mai sus. Maddox și Lilie disting însă între două dimensiuni ale dezbaterii publice, între două seturi de probleme: chestiunile legate de extinderea libertăților personale (accesul la informații, dreptul la avort etc.) și chestiunile legate de intervenția statului în economie. Fiecare tip de om politic are o atitudine: pro sau contra libertăților personale extinse, pro sau contra intervenției statului în economie. Introducerea celor două seturi de probleme și a atitudinilor în problemele respective complică desigur tabelul.

Procedez pas cu pas. Fac abstracție de termenii „liberal“, „libertarian“, „populist“, „conservator“; pun doar niște semne convenționale în locul lor. Mai întâi introduc un rând și o coloană suplimentare. Pun doar + și -, după caz, în patru casete ale tabelului și las colțul din stânga sus liber.

Mai complicat este cu dimensiunile ca atare. Textul este lung și așa avea imediat probleme cu spațiul în pagină. Trag concluzia că, într-o primă fază, este suficient să scriu un 1, respectiv 2. Dar unde? Trebuie ca 1 să ocupe două coloane, iar 2 două rânduri! Asta și fac; pun 1 în două casete și 2 în două casete.

¹⁰⁵Pagina de web a programului este <http://g32.org>.

¹⁰⁶William S. Maddox și Stuart A. Lilie, *Beyond Liberal and Conservative: Reassessing the political spectrum* (Washington: Cato Institute, 1984), p.5.


```
\begin{tabular}{lccc}
& & 1 & 1 \\
& & + & - \\
2 & + & L1 & L2 \\
2 & - & P & C
\end{tabular}
```

		1	1
		+	-
2	+	L1	L2
2	-	P	C

Rezultatul este o schemă a tabelului final. Pe baza acestei scheme vom dezvolta tabelul final.

B.4.1.1 Tabelul complet și trimerile la tabele

- 5 De multe ori este nevoie să ne referim la un tabel care nu se află în imediata apropiere. În acest caz tabelului trebuie să-i dăm un nume și trebuie să declarăm o etichetă care să ne permită să folosim mecanismul trimerilor din \LaTeX .

Până acum tabelele au fost moduri de a diviza convenabil spațiul disponibil pe o pagină. Acum tabelul trebuie încadrat într-un mediu \LaTeX care-i permite să *plutească* în pagină.¹⁰⁷

Poziția corpului plutitor o alege, în cele din urmă, \LaTeX . Avantajul din punctul de vedere al autoarei sau autorului este că se poate referi, precum în cazul nostru, la „tipuri de ideologii“ adăugând și

15 „vezi tabelul B.1“.

Cum am obținut tot acest efect? Planul a fost foarte simplu: Am luat schema de tabel deja creată și am introdus-o într-un mediu `table`. În mediul `table` se poate da o comandă `\caption{}` pentru titlul tabelului. Eticheta este atașată tot mediului `table`. Trimerirea

20 la tabel se face cu obișnuita comandă `\ref{}`.

```
1 \newcommand{\mc}[3]{\multicolumn{#1}{#2}{#3}}
2 \begin{table}[ht]
3 \centering
4 \begin{tabular}[c]{lccc}
5 & & & & & & & & & \\
6 & & & & & & & & & \\
7 & & & & & & & & & \\
8 & & & & & & & & & \\
9 \cline{3-4}
10 & & & & & & & & & \\
11 Extinderea & & & & & & & & & \\
12 libertăților & & & & & & & & & \\
13 personale & & & & & & & & & \\
14 \cline{3-4}
15 \end{tabular}
16 \caption{Tipuri de ideologii după Maddox și Lilie}
```

¹⁰⁷ Termenul tehnic în engleză este cel de *float*. Nu doar imaginile plutesc în pagini. Notele marginale sunt, de pildă, și ele structuri plutitoare.

```
17 \label{TipuriDeIdeologii}  
18 \end{table}
```

Comanda `\multicolumn{}{}{}` merită o atenție specială. Pentru că numele ei este lung am redenumit-o mc (pe rândul 1 din fragmentul de sursă `LATEX` de mai sus). Primul ei argument specifică numărul de coloane. Acest număr poate fi și 1! Al doilea argument arată cum este aliniat textul. Al treilea argument este reprezentat de text ca atare. În rezumat, este o comandă care spune pe câte coloane este scris un text în tabel și cum este aliniat. 5

Dacă după litera care specifică alinierea punem o bară verticală¹⁰⁸, atunci în tabel, la capătul coloanei(coloanelor) respective apare o linie verticală. 10

Liniile orizontale sunt trasate cu `\cline{}`; în argumentul acestei comenzi trebuie specificat de la ce coloană până la ce coloană se trage linia. Comanda `\hline` trage o linie orizontală de-a lungul întregului tabel.

		Intervenția statului în economie	
		pro	contra
Extinderea libertăților personale	pro	liberalism	libertarianism
	contra	populism	conservatorism

Tabelul B.1: Tipuri de ideologii după Maddox și Lilie

Pentru a obține tabelul B.1 nu a mai fost nevoie decât să refac conținutul miezului inițial al tabelului. 15

Dacă numerotarea tabelelor nu începe de la 1, dați comanda `\setcounter{table}{0}` înainte de primul tabel.

B.4.1.2 Tabelele și aranjarea textului în pagină

Tabelele sunt folosite intens și pentru a crea aranjamente (uneori destul de complicate) ale textului în pagină. De exemplu, pentru a aranja lista de coduri și simboluri de la pagina 247 am folosit un tabel. Redăm aici doar comenzile pentru un rând, fără a include codurile sau simbolurile ca atare. 20

¹⁰⁸În limbajul Vim este vorba despre `<char-124>`.

```

1 \begin{longtable}[c]{lllll}
2 ---\textless{}\dots\textgreater{}\&
3 ---\textless{}\dots\textgreater{}\&
4 ---\textless{}\dots\textgreater{}\&
5 ---\textless{}\dots\textgreater{}\&
6 ---\textless{}\dots\textgreater{}\&
7 .....
8 \end{longtable}

```

Scopul este dispunerea textului în așa fel încât să fie cât mai lizibil. Am folosit mediul `longtable` pentru ca \LaTeX să poată decupa tabelul în porțiuni care sunt afișate pe pagini diferite. Opțiunea `c` din primul rând de comenzi cere sistemului să centreze întregul tabel în cadrul paginii.

B.4.2 Câteva idei simple despre formule

\LaTeX este îndeosebi faimos pentru capacitatea sa de a genera formule matematice. În secțiunea B.1.3.4.3, referitoare la modul matematic, am arătat deja cum ne putem sluji de modul matematic în rândurile de text. Multe eseuri filosofice conțin formule în limbajul simbolic al logicii. Este, de aceea, util să trecem în revistă modul simbolistica de bază a logicii poate fi realizată în \LaTeX .

Negația : $\$ \neg \{p\} \$$

O notăție pentru conjuncție: $\$ p \& \{q\} \$$

Altă notăție pentru

conjuncție: $\$ p \wedge \{q\} \$$

Implicația: $\$ p \rightarrow \{q\} \$$

Echivalența: $\$ p \equiv \{q\} \$$

Negația : $\neg p$

O notăție pentru conjuncție:

$p \& q$

Altă notăție pentru conjuncție:

$p \wedge q$

Implicația: $p \rightarrow q$

Echivalența: $p \equiv q$

Acestea sunt doar câteva dintre posibilele soluții în cazul logicii propozițiilor. Trebuie consultate tabelele cu simboluri matematice din documentația \LaTeX pentru a descoperi întregul evantai de simboluri.

În cazul logicii predicatelor putem folosi:

Cuantificare existențială: $\$ (\exists x) (P x \wedge Q x) \$$

Cuantificare universală: $\$ (\forall x) (P x \rightarrow Q x) \$$

Cuantificare existențială: $(\exists x)(P x \wedge Q x)$

Cuantificare universală: $(\forall x)(P x \rightarrow Q x)$

Operațiile cu mulțimi și relațiile dintre mulțimi sunt și ele ușor de reprezentat cu ajutorul \LaTeX .

x aparține mulțimii X :	$x \in X$
Negația apartenenței:	$x \notin X$
Incluziunea:	$Y \subset X$
Intersecția:	$X \cap Z$
Reuniunea:	$X \cup Z$
Complementara:	$\sim X$

B.4.2.1 Formulele din cărțile reale

Rezultatele matematice sunt exprimate ca formule în limbaj simbolic și, de regulă, sunt scrise pe rânduri separate. Oricum, atunci când cităm formule din cărțile altora, trebuie să le punem pe un rând separat. 5

De exemplu, Adrian Miroiu, într-un studiu în care dezvoltă ideea sa a lumilor în interiorul lumilor, introduce un prim rezultat general sub forma următorului enunț:¹⁰⁹

$$\vdash_{LK} \varphi \text{ ddacă } \models_{LK} \varphi$$

Iată cum am codificat în \LaTeX enunțul din cartea lui Adrian Miroiu:

```
1 \begin{displaymath}
2 \mathrm{\vdash_{LK}\varphi \text{ ddacă } \models_{LK}\varphi}
3 \models_{LK}\varphi
4 \end{displaymath}
```

Codificarea \LaTeX este destul de transparentă pentru cei care știu logică. S-ar putea spune chiar că oferă un plus de explicații. Trebuie totuși făcute câteva comentarii. 10

Prima observație este aceea că trebuie să tratăm modul matematic ca un loc în care n-are sens să venim cu obiceiurile de a scrie în mod text. În primul rând, în mod automat, \LaTeX scrie cursiv textul matematic. De aici necesitatea comenzii `\mathrm{}`, care-i spune sistemului să scrie cu tipul de literă roman. 15

litere
grecești în
modul
matematic

În al doilea rând, folosirea literelor în modul matematic are particularitățile sale.¹¹⁰ Literele grecești, de pildă, se obțin cu ajutorul unor comenzi speciale. Am folosit `\varphi` și nu `\phi` pentru că aceasta era varianta lui φ folosită în textul lui Adrian Miroiu. 20

În al treilea rând, trebuie observat rolul spațierii și alegerii dimensiunii literelor în modul matematic. Barele oblice inverse urmate

¹⁰⁹ Adrian Miroiu, *Constructe formale* (București: Editura Trei, 2000), p.16.
¹¹⁰ Pentru literele românești în modul matematic vezi §B.4.2.2.

de un spațiu sunt comenzi! Ele cer introducerea unui spațiu, menit să elimine eventuale ambiguități. De asemenea, o pereche de acolade (fără nimica între ele) forțează mărirea spațiului alb. De aici lipsa lor după `\vdash` sau `\models`. Cu acolade după comenzi, rezultatul

5 ar fi $\vdash_{LK} \varphi$ sau $\models_{LK} \varphi$.

Declarația `\scriptscriptstyle` corectează dimensiunea indicelui, reducând-o la dimensiunea unui indice la indice. Observați, în acest sens, diferența dintre $\vdash_{LK} \varphi$ și $\vdash_{LK} \varphi$.

Un alt exemplu, extras dintr-o carte a lui Mircea Dumitru, pune

10 și el probleme interesante de tehnoredactare.¹¹¹ Fragmentul citat mai jos este o parte din definiția noțiunii de satisfacere în logica modală a propozițiilor (LLMP). Iată fragmentul:

pentru fiecare fbf Φ în LLMP,

$$(\mathcal{M}, w) \models \Diamond \Phi \text{ dacă } (\exists u \in W_{\mathcal{M}})(R_{\mathcal{M}} w u \text{ și } (\mathcal{M}, u) \models \Phi)$$

Prima problemă a codificării este scrierea pe mai multe rânduri. Ar trebui să construim un tabel! A se vedea acest lucru, pentru

15 modul matematic, în §B.4.2.4. Formula în sine este ușor de înțeles pentru cei care au citit un curs elementar de semantică a logicilor modale. Chiar dacă nu o înțelegeți, problemele de tehnoredactare sunt interesante în sine.

Există o simetrie între cele două rânduri citate. Primul este scris

20 în modul text, dar are o literă grecească realizată în mod matematic. Al doilea rând este scris în modul matematic, dar are două inserții în mod text după cum se vede din rândurile de cod de mai jos.

```
1 pentru fiecare fbf  $\Phi$  în LLMP,
2 \begin{displaymath}
3 (\mathcal{M}, w) \models \Diamond \Phi \text{ dacă }
4 (\exists u \in W_{\mathcal{M}}) (R_{\mathcal{M}} w u \text{ și }
5 (\mathcal{M}, u) \models \Phi)
6 \end{displaymath}
```

text în
modul
matematic

Comanda `\text{}` cu ajutorul căreia am inserat text în modul matematic nu este accesibilă dacă ați încărcat doar nucleul sistemului

25 \LaTeX . Trebuie să încărcăți pachetul `amstext`.¹¹²

¹¹¹Mircea Dumitru, *Modalitate și incompletitudine* (București: Paideia, 2001), pp.22-23.

¹¹²Pachetul `amstext` ar fi încărcat automat dacă aș folosi \LaTeX . n-are însă rost să încarc artileria grea de la American Mathematical Society în această carte. Pentru o explicație mai detaliată a comenzii `\text{}` vezi cartea lui Paul A. Blaga și Horia F. Pop[4, p.156].

Oricât ar putea părea de curios problema cea mare este litera ș de pe rândul 4 din codul de mai sus. Obținerea ei cu ajutorul comenzii `\text{}` este mai ușoară decât realizarea ei în modul matematic pur.

Deși exemplele sunt simple în conținutul lor, tehnoredactarea lor pune același tip de probleme pe care le-ar formaulele cu un conținut mai dificil din cărțile citate. Pentru listele de simboluri care pot fi utilizate trebuie consultată documentația.¹¹³

B.4.2.2 Modul matematic și literele românești

Punerea de diacritice pe litere în modul matematic diferă de operația similară din modul text. Dacă scriu un ă în mod matematic, sistemul traduce acest lucru prin comanda `\u{a}`, care este inacceptabilă în mod matematic. Trebuie scris `\breve{a}` în mod matematic.

Iată literele românești în mod matematic pur (fără pachetul `amstext`):

`\breve{a}`\$ `\breve{A}`\$\\
`\hat{a}`\$ `\hat{A}`\$\\
`\hat{\imath}`\$ `\hat{I}`\$\\
`\stackrel{\textstyle}{s}`{,}\$
`\stackrel{\textstyle}{S}`{,}\$\\
`\stackrel{\textstyle}{t}`{,}\$
`\stackrel{\textstyle}{T}`{,}\$

ă	Ă
â	Â
î	Î
s	S
t	T

Comenzile pentru ș și ț în mod matematic sunt meșterite cu ajutorul comenzii `\stackrel{}{}{}`. Ea este larg folosită pentru a produce simboluri precum $\stackrel{df}{=}$ (cu comanda `\stackrel{df}{=}`\$).

Exemplele de mai sus nu au probabil mare valoare practică, dar ele ilustrează posibilitatea de a crea în \LaTeX simbolurile de care avem nevoie.

B.4.2.3 Numerotarea formulelor

Putem pune pe rânduri separate formule, fără să le numerotăm. Este păcat însă să nu ne folosim de posibilitatea de a numerota formulele. Putem apoi folosi mecanismul standard din \LaTeX pentru a ne referi la o formulă.

¹¹³Eu am folosit în special Scott Pakin „The Comprehensive \LaTeX Symbol List“ (v. fișierul `symbols.dvi` în dosarul `comprehensive` din documentația \LaTeX).

Deși pare complicată, axioma Nicod–Łukasiewicz pentru logica propozițiilor(B.1) nu este greu de scris în \LaTeX .

$$[p/(q/r)]/\left([s/(s/s)]/\{(s/q)/[(p/s)/(p/s)]\}\right) \quad (\text{B.1})$$

Propoziția compusă p/q este falsă dacă ambele variabile propoziționale iau valoarea *adevărat*; altfel este adevărată. Codul \LaTeX folosit la scrierea mediului în care este plasată axioma este următorul:

```
1 \begin{equation}
2 [p/(q/r)]/\bigg([s/(s/s)]/\{(s/q)/[(p/s)/(p/s)]\}\bigg)
3 \label{NicodŁukasiewicz}
4 \end{equation}
```

Referirile la formulele numerotate se fac cu ajutorul comenzii `\ref{}` care ia drept argument eticheta declarată în cadrul mediului în care este scrisă formula.

10 B.4.2.4 Tabele care conțin formule

Tabelele care conțin text se construiesc în mediul `tabular`. Ar fi posibil să introducem simboluri sau formule matematice și-n aceste tabele, prin trecerea în porțiunea respectivă în modul `matematic`. Recomandabil este însă să folosim pentru tabelele cu formule mediul

15 `array`.

Voi începe cu un tip de tabel familiar oricui începe să învețe logică: matricile conectorilor propoziționali.

```
\begin{displaymath}
\begin{array}{llc}
p \ \& \ q \ \& \ p \rightarrow q \\
\hline
1 \ \& \ 1 \ \& \ 1 \\
1 \ \& \ 0 \ \& \ 0 \\
0 \ \& \ 1 \ \& \ 1 \\
0 \ \& \ 0 \ \& \ 1
\end{array}
\end{displaymath}
```

p	q	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

Pentru un tabel asemenea celui de mai sus, scrieți mai întâi comenzile pentru mediul care afișează formule nenumerate. Procedeți apoi ca și-n cazul mediului `tabular`, dar folosiți mediul `array`, în care puteți introduce formule.

Tabelele cu formule sunt evident mult mai utile dacă ne putem referi la ele „de la distanță”. De exemplu, vreau să vă dau un exemplu

de demonstrare a unei tautologii din logica propozițiilor. Atunci voi trimite la tabelul B.2.

p	q	$p \rightarrow q$	$p \wedge (p \rightarrow q)$	$(p \wedge (p \rightarrow q)) \rightarrow q$
1	1	1	1	1
1	0	0	0	1
0	1	1	0	1
0	0	1	0	1

Tabelul B.2: Demonstrarea unei tautologii

Sursa L^AT_EX a tabelului B.2 sper că este suficient de clară pentru a înțelege modul în care este construit tabelul. Am evitat în mod special utilizarea & drept conectiv pentru conjuncție pentru a nu genera confuzii. Simbolul rezervat & este separatorul coloanelor, nu și nume pentru o comandă de generare a simbolului conectorului conjuncției.

```
1 \begin{table}[htb]
2 \centering
3 \begin{displaymath}
4 \begin{array}{llccc}
5 p & \& q & \& p \rightarrow q & \& p \wedge (p \rightarrow q) & \& (p \wedge (p \rightarrow q)) \rightarrow q \\
6 \hline
7 1 & \& 1 & \& 1 & \& 1 & \& 1 \\
8 1 & \& 0 & \& 0 & \& 0 & \& 1 \\
9 0 & \& 1 & \& 1 & \& 0 & \& 1 \\
10 0 & \& 0 & \& 1 & \& 0 & \& 1
11 \end{array}
12 \end{displaymath}
13 \caption{Demonstrarea unei tautologii}
14 \label{DemonstrareaUneiTautologii}
15 \end{table}
```

Puteți localiza poziția unui tabel în cadrul sursei L^AT_EX cu ajutorul T_EXnicCenter. Puteți, de asemenea, produce o listă a tabelelor în documentul final, plasând comanda \listoftables după comanda \tableofcontents din fișierul principal al proiectului.

Este necesar să comentăm este necesar puțin folosirea liniilor în tabele. Recomandarea noastră este de a folosi linii doar pentru a face tabelul clar. Liniile superflue sau cu caracter pur decorativ nu-și au locul într-o lucrare academică.

O linie sub capul de tabel este, de multe ori, suficientă. Dacă tabelul conține elemente de sinteză în partea de jos a coloanelor, atunci

o linie simplă sau dublă este necesară deasupra ultimului rând. Ea face tabelul mai clar. Abuzul de linii este total nerecomandabil.

Crearea de tabele poate fi un lucru extrem de pretențios. Adesea ele sunt prea late sau prea lungi și așa mai departe. Prezentarea noastră a fost doar una sumară. Trebuie să citiți o carte despre \LaTeX care acordă mai mult spațiu tabelelor.¹¹⁴ Foarte utilă este și documentația inclusă în distribuția \MikTeX .

B.4.2.5 Formulele ca tabele

Am pregătit recent un text în care voiam să citez o funcție pe care o discută Kripke.¹¹⁵ Funcția, pe care Kripke o numește „quus” (simbolic \oplus), generează două tipuri de valori: suma valorilor argumentelor sale și o valoare constantă. Condiția pentru a genera primul tip de rezultate este ca valoarea oricăruia dintre cele două argumente ale funcției să fie mai mică de 57. Altfel, este generată constanta.

Pentru a descrie simbolic funcția *quus* Kripke folosește o notație cu două semne de egalitate și nici o acoladă. Practica uzuală este însă diferită și o voi respecta aici.¹¹⁶

Chiar dacă descrierea simbolică nu este complicată, trebuie să gândim tot din perspectiva proiectului. Obiectivele noastre sunt să prezentăm:

unu numele funcției și argumentele sale;

doi valoarea generată când este adevărată prima condiție;

trei valoarea generată când este adevărată a doua condiție.

Pentru a atinge primul obiectiv este suficient să scriem un rând de text matematic. Al doilea și al treilea obiectiv presupun folosirea unui tabel (cu două coloane și două rânduri).

Construim separat elementele formulei B.2 ca-n exemplele care urmează. Ca de obicei, este important să executăm totul pas cu pas. Scriem mai întâi rândul de text matematic.

```
\begin{displaymath}
x \oplus y =
\end{displaymath}
```

$$x \oplus y =$$

¹¹⁴A se vedea, în limba română, cartea lui Paul A. Blaga și Horia F. Pop[4, pp.50-55, 212-219].

¹¹⁵În Saul A. Kripke, *Wittgenstein on Rules and Private Language* (Oxford: Blackwell, 1982), p.9.

¹¹⁶Pentru un model care respectă standardele uzuale v. Lamport[21, p.47].

Elaborăm apoi tabelul cu elementele generate de către funcție, în cele două condiții specificate mai sus.

```
\begin{displaymath}
\begin{array}{ll}
x+y, & \text{\text{dacă}}\ x,y<57\\
5 & \text{\text{altfel}}
\end{array}
\end{displaymath}
```

$x + y,$	$\text{dacă } x, y < 57$
5	altfel

Testăm posibilitatea de a pune tabelul între acolade.

```
\begin{displaymath}
\left\{
\begin{array}{ll}
x+y, & \text{\text{dacă}}\ x,y<57\\
5 & \text{\text{altfel}}
\end{array}
\right.
\end{displaymath}
```

$\left\{ \begin{array}{ll} x + y, & \text{dacă } x, y < 57 \\ 5 & \text{altfel} \end{array} \right\}$

5

Acum nu este greu să obținem forma finală, corectând eventuale inexactități, și eliminând acolada din dreapta.

$$x \oplus y = \left\{ \begin{array}{ll} x + y, & \text{dacă } x, y < 57 \\ 5 & \text{altfel} \end{array} \right. \tag{B.2}$$

Perechea de comenzi `\left \right` permite crearea de perechi de paranteze. Felul în care arată paranteza depinde de felul în care completăm numele comenzii. Dacă punem `{` apare acolada din stânga. Putem pune și o paranteză rotundă, una pătrată sau o bară verticală (pentru matrici). Dacă vrem ca paranteza să fie invizibilă, atunci punem un punct după numele comenzii. Iată acum și forma finală a codului \LaTeX pentru descrierea funcției lui Kripke.

10

```
1 \begin{equation}
2 x \oplus y = \left\{
3 \begin{array}{ll}
4 x+y, & \text{\text{dacă}}\ x,y<57 \\
5 5 & \text{\text{altfel}}
6 \end{array}
7 \right.
8 \label{quus}
9 \end{equation}
```

În ciuda aparentei simplități a codului \LaTeX , dacă încercați să-l construiți fără nici un plan, s-ar putea să vă treziți într-un noian de erori. Orice tabel, oricât de simplu, trebuie proiectat atent.

15

B.4.3 Tehnicile avansate de scriere matematică

Scrierea surselor \LaTeX pentru modul matematic este un lucru mult mai pretențios decât pregătirea părții care este în mod text. Din fericire, cărțile de logică sau de filosofia științei, cele care folosesc de obicei cel mai intens limbajele simbolice, nu conțin la tot pasul sisteme de ecuații complicate.

În cazul când vreți să scrieți sisteme de ecuații sau să grupați grafic rânduri cu formule, trebuie să folosiți mediul `eqnarray`.

Dacă doriți să obțineți rezultate și mai sofisticate, trebuie să recurgeți la `amsmath` și alte pachete dedicate special matematicii.

Marea majoritate a introducerilor în \LaTeX sunt scrise din perspectiva utilizării sistemului pentru a tehnoredacta studii și cărți de matematică și informatică. Nu veți duce lipsă de surse de documentare dacă vă interesează tehnicile avansate de elaborare a textelor care conțin formule complicate.¹¹⁷

B.5 Indexarea electronică

\LaTeX folosește un program special pentru a genera indici. Folosirea acestui compilator de indici este mai simplă decât în cazul \BibTeX .

Ca și-n cazul \BibTeX , mediul integrat `TEXnicCenter` are, la proprietățile proiectului, o opțiune pentru folosirea programului `MakeIndex`¹¹⁸. Această opțiune trebuie bifată, dacă vreți să generați un indice.

Studiați, de asemenea, `Define Output Profiles...` din meniul `Build` al `TEXnicCenter` și veți vedea cum este apelat `makeindex.exe` în mediul integrat.

În preambulul documentului principal al proiectului trebuie să introduceți două comenzi \LaTeX :

```
1 \usepackage{makeidx}
2 \makeindex
```

La sfârșitul corpului documentului principal, după comenzile pentru bibliografie, introduceți comanda `\printindex`.

¹¹⁷ În limba română puteți consulta cartea lui Paul A. Blaga și Horia F. Pop[4].

¹¹⁸ Autorul programul *MakeIndex* este Pehong Chen. Pentru detalii referitoare la program și utilizarea lui, a se vedea Leslie Lamport, *MakeIndex: An Index Processor for \LaTeX* , `\texmf\doc\makeindex\makeindex.dvi` în documentația Mik \TeX .

Marele avantaj al indexării în \LaTeX este posibilitatea de a indexa din mers. N-aş recomanda însă acest fel de a indexa. Pentru a afla principiile indexării studiaţi secţiunea 5.4.1.

Comanda de bază este `\index{}`. În argumentul ei se plasează un descriptor. Comanda aceasta trebuie folosită precum comanda `\label{}`; ea trebuie pusă acolo unde vrem ca indicele de la sfârşit să ne trimită în text.

Creaţi un proiect test şi indexaţi în felul indicat mai sus. Vedeţi ce apare în fişierul `dvi`. Examinaţi, de asemenea, fişierul de tip `ind`, care este generat automat de *MakeIndex*. Nu este greu să vă daţi seama c-aţi putea crea acest fişier şi manual. Atâta timp cât utilizaţi *MakeIndex*, nu operaţi schimbări în acest fişier. Ele vor fi distruse la orice recompilare. Modificaţi comenzile din surse.

Puteţi folosi subdescriptori după modelul de mai jos:

```
1 \index{indici!generarea manuală}
2 \index{indici!generarea lor cu MakeIndex}
```

Descriptorul este separat de subdescriptor cu ajutorul semnului exclamării.

Pentru a realiza trimiteri de la un descriptor la altul în cadrul listei de indici folosiţi modelul următor:

```
1 \index{editor de texte|see{Vim}}
```

Bara verticală precede trimiterea. Pachetul `babel` va traduce pe `see`.

Folosiţi capacităţile de căutare cu ajutorul expresiilor regulate ale `TEXnicCenter` pentru a unifica intrările. De asemenea, trebuie să compilaţi cel puţin de două ori documentul pentru ca să apară modificările operate în indice.¹¹⁹

Pentru a imita notarea tradiţională a descriptorilor pe marginea paginii folosiţi pachetul `showidx`. Pachetul are un rol pur auxiliar. Nu eliminaţi *bad box*-urile generate de notele marginale introduse de `showidx`. Opriţi utilizarea acestui pachet după ce aţi terminat indexarea.

¹¹⁹Prima oară este modificat doar fişierul de tip `ind`. Abia a doua oară se modifică şi documentul final. Dacă aveţi probleme, citiţi mesajele compilatorului.

Anexa C

Pensula electronică

`LATEX` are propriile sale comenzi și medii pentru realizarea de diagrame și desene cu caracter tehnic. Prezentarea lor ar depăși cu mult
5 ambițiile foarte reduse ale acestei anexe.

Eseurile filosofice nu abundă în diagrame și nici în desene sau imagini. Câteva noțiuni elementare privitoare la imaginile computerizate și integrarea lor în sursele `LATEX` sunt suficiente pentru persoanele care scriu un eseu filosofic.

10 C.1 Imaginile

În anexa B ne-am referit adesea, în mod metaforic, la „literele de plumb” ale meșterului tipograf. Computerul n-are litere de plumb. Literele de pe ecran sunt desenate din niște puncte denumite „pixeli”¹; punctele acestea pot fi albe, negre sau colorate.

15 Dacă imprimați pe un printer cu jet de cerneală, literele de pe foaia de hârtie sunt compuse din picături de cerneală extrem de mici.

Atunci când tehnoredactăm un text în `LATEX` nu ne interesează punctele de pe ecran sau de pe foaia de hârtie. Gândim din perspectiva
20 cutiilor cu litere plasate pe ecran sau pe foaie, nu gândim la nivelul pixelilor. Dacă vrem să înțelegem însă secretele imaginilor generate de către computer, trebuie să ne îndreptăm atenția către aceste puncte minuscule.

O primă strategie de realizare a imaginilor pe ecran este aceea
25 de a nota într-un fișier informațiile despre pixeli, în așa fel încât

¹Denumirea vine de la *picture element* (element al unei imagini).

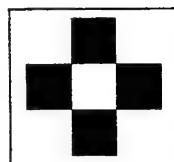
să fim în stare să reconstituim imaginea pe ecran cu ajutorul unui program de vizualizare.

Puteti face, folosind Vim ca pe un editor de coduri hex, precum și un program de vizualizare ca IrfanView,² câteva experimente cu un fișier ce stochează imagini.

Deschideți IrfanView și mergeți la rubrica **Create New (empty) image** din meniul **Image**. Creați o imagine de 3×3 pixeli și cu 24BPP.³ Salvați fișierul nou creat dându-i o denumire urmată de extensia **bmp**.

Deschideți fișierul nou creat cu Vim și treceți la editarea în hex. Modificați fișierul după modelul de mai jos, schimbând doar ultimele trei rânduri, fără a altera ultimele patru cifre hex. Treceți din nou în modul normal, salvați și închideți.

```
424d 5a00 0000 0000 0000 3600 0000 2800
0000 0300 0000 0300 0000 0100 1800 0000
0000 2400 0000 9c0e 0000 9c0e 0000 0000
0000 0000 0000 ffff ff00 0000 ffff ff00
0000 0000 00ff ffff 0000 0000 0000 ffff
ff00 0000 ffff ff00 0000 0d0a
```



Atenție! Manipularea efectuată nu va scăpa neobservată cuiva competent (unele caracteristici ale imaginii, specificate pe rândul al treilea, nu mai corespund), dar fișierul poate fi vizualizat cu IrfanView. Imaginea, mărită suficient cu ajutorul IrfanView,⁴ arată ca un fragment dintr-o tablă de șah, cu pătrate albe și pătrate negre. Pătratele albe au culoarea specificată cu ajutorul șirurilor **ffffff**.

Schimbați un grup **ffffff** în **ff0000**. Nu uitați să treceți din hex în modul normal Vim și să salvați schimbarea. Pixelul a devenit albastru în programul de vizualizare, după ce ați reîmprospătat⁵ imaginea. Schimbați pe **ffffff** în **00ff00** și pixelul devine verde. Schimbați pe **ffffff** în **0000ff** și pixelul devine roșu.

Fișierele create în modul descris mai sus, conțin o hartă a pixelilor. Nu este de mirare că, atunci când mărim o astfel de imagine, lucrurile arată precum în figura C.1.

Atunci când schimbăm dimensiunile unei imagini bazate pe o hartă a pixelilor, în mod inevitabil programul de vizualizare n-are

²Programul IrfanView poate fi descărcat gratuit de la adresa de Internet <<http://www.irfanview.com>> și este un program de vizualizare de imagini foarte popular.

³Practic, 24BPP înseamnă că fiecărui pixel îi corespund patru perechi de două cifre hex. Primele trei perechi sunt folosite pentru informații despre culorile albastru, verde și roșu.

⁴Apăsați tasta F.

⁵În cazul Vim și IrfanView trebuie să închideți și să redeschideți fișierele.

Eseul filosofic

Figura C.1: O imagine care folosește o hartă a pixelilor mărită

informații suficiente decât pentru niște pete de culoare mai mari. Imaginea capătă un aer de pictură impresionistă.

C.1.1 Programul Gimp

Vim este un excelent editor de fișiere de tip text, dar n-ar putea fi folosit pentru editarea de imagini. Într-un fel, exemplul cu care am început secțiunea §C.1 ne arată de ce acest lucru este adevărat.

Programul pe care l-aș recomanda în mod deosebit pentru prelucrarea hărților de pixeli este Gimp.⁶ Gimp poate deschide inclusiv mica imagine creată prin manipularea codurilor hex (vezi pagina 278, rândul 12).

Posibilitățile Gimp sunt numeroase, dar descrierea lor nu constituie obiectivul anexei de față. Multe dintre acțiunile pe care le puteți întreprinde cu Gimp sunt evidente din denumirea meniurilor contextuale ale ferestrelor programului.

O singură observație ar fi de făcut aici. Ar fi inutil să încercați să schimbați o imagine de 3×3 pixeli cu pensulele Gimp, oricât de mici le-ați face. Selectați zona (practic pixelii) pe care vreți să o modificați și umpleți-o cu culoarea dorită.

C.1.2 Limbajul PostScript

PostScript este un limbaj creat de către firma Adobe. A fost inițial proiectat ca limbaj special pentru printere. Variantele ulterioare au adăugat elementele necesare pentru lucrul și din perspectiva ecranului.

⁶Vizitați <<http://www.gimp.org>> pentru a afla cum puteți descărca versiunea Gimp pentru Windows. Sub Linux, Gimp este programul standard de prelucrare a imaginilor și îl conține orice distribuție de Linux. Gimp este o prescurtare de la „GNU image manipulation program“. A fost creat de Spencer Kimball și Peter Mattis. Ca orice program GNU, cu sursă deschisă, include contribuții a numeroși programatori. Chiar și versiunea sub Windows este uimitor de stabilă și de eficientă.

Sursele scrise în PostScript sunt fișiere de tip text. Vim este perfect capabil să creeze fișiere PostScript. Aceste fișiere au extensia ps. Pentru interpretarea fișierelor PostScript este nevoie de un program precum Ghostscript.

Pentru a vă da seama care este diferența dintre grafica PostScript și cea bazată pe hărți ale pixelilor, cel mai simplu lucru este să experimentați puțin cu un fișier PostScript. Creați un fișier phi.ps cu următorul conținut:⁷

```
1 %!PS
2 %%BoundingBox: 18 28 124 180
3 /cm { 28.35 mul } def % definiția centimetrului
4 %%EndProlog
5 1 cm 6 cm moveto      % punctul din care va fi desenată figura
6 1 cm 3.5 cm lineto    % marcarea unei căi
7 4 cm 3.5 cm lineto    % marcarea mijlocului lui phi
8 4 cm 6 cm lineto      % începem bucla lui phi
9 2.5 cm 6 cm lineto    % ne întoarcem către mijloc
10 2.5 cm 1 cm lineto   % ultima cale marcată
11 0.7 cm setlinewidth  % grosimea liniei pe calea trasată
12 2 setlinejoin        % rotunjirea colțurilor
13 stroke               % punerea de cerneală pe calea trasată
14 showpage             % afișarea paginii
```

Rezultatul, vizualizat cu Ghostscript este o versiune stilizată a literei grecești phi.



Persoanele familiarizate cu notația poloneză din logica simbolică pot înțelege mai ușor sintaxa PostScript. Limbajul creat de către firma Adobe folosește notația poloneză inversă.⁸

Primul rând din orice program PostScript începe cu %!PS. Sar peste cel de al doilea rând. Voi reveni la el mai târziu. Al treilea rând conține o definiție a centimetrului. Definiția este necesară pentru că PostScript folosește ca unică unitate de măsură punctele PostScript. 267 de puncte PostScript sunt egale cu aproximativ 268 de puncte tipografice.

⁷Drept ghid pentru limbajul PostScript am folosit McGilton și Campione[26].

⁸A se vedea și limbajul bst în secțiunea B.2.3.

Rândurile 5-10 cuprind comenzile de trasare a unei căi. Ele nu desenează nimic. Dacă puneiți semnul comentariului % în fața penultimului rând, veți vedea că nu se desenează absolut nimic. Ideea de bază este că se trasează o cale și apoi se așterne cerneala pe calea respectivă. Restul se poate deduce din comentariile incluse în program.

Măriți imaginea literei phi. Nu se produce nici un fel de spargere în puncte a literei. Ea este perfect scalabilă.

Explicația de principiu a fenomenului de mai sus este simplă.

Dacă imaginea se mărește, programul retrasează calea, în funcție de procentul cu care cresc dimensiunile imaginii. Cerneala este adăugată abia ulterior. Interpretorul de PostScript nu depinde în acest caz de datele privitoare la pixeli. Grafica de acest tip se numește „grafică vectorială” și contrastează cu „bitmap”-urile (cu hărțile de pixeli).

Acum putem explica și cel de al doilea rând din programul PostScript. Acest rând declară o ramă, o cutie în jurul imaginii. Este aproape inutil să spun că această cutie din jurul imaginii joacă un rol deosebit de important în integrarea imaginii PostScript în L^AT_EX.

Ghostview poate genera automat cutia. Ștergeți rândul 2 din phi.txt și deschideți fișierul cu Ghostview. Creați un fișier PostScript încapsulat cu ajutorul dialogului PS to EPS din meniul File. Salvați-l cu numele phi.eps și citiți-l ca text cu ajutorul Vim. Veți găsi cutia creată în mod automat. O puteți vizualiza în Ghostview cu ajutorul opțiunii Show Bounding Box.

C.1.2.1 Prelucrarea fișierelor PostScript în Gimp

Gimp este preponderent orientat către grafica hărților de pixeli. Este însă perfect capabil să deschidă un fișier PostScript. Trebuie însă să aveți instalat Ghostscript pe computer. Gimp folosește Ghostscript pentru a lucra cu fișiere PostScript.

Gimp poate, de pildă, deschide fără probleme fișierul phi.ps, cu sau fără cutia delimitatoare.

Fiți atente și atenți la faptul că Gimp, dacă modificați fișierul PostScript, se va folosi de posibilitatea de a construi hărți de pixeli în PostScript! Pe lângă faptul că fișierul va fi de needitat cu mâna, imaginea va avea probleme la scalare ca orice bitmap.

Dacă aveți deja creat un fișier PostScript, lucrul cel mai bun pe care-l poate face Gimp este să-l transforme într-un fișier jpg.

Ciclul optim de lucru cu Gimp este în trei timpi. În primul timp,

creați un fișier `xcf` (formatul nativ al Gimp) și prelucrați acest fișier. În al doilea timp, transformați fișierul `xcf` într-un fișier PostScript cu *BoundingBox*. În al treilea timp, transformați acest fișier într-un fișier `jpg`.⁹

În sfârșit, nu uitați că trecerea de la fișierul `jpg` la cel `xcf` nu este, de fapt, posibilă. Fișierul `jpg` comprimă imaginea creată. Faceți toate prelucrările în formatul nativ Gimp.

C.2 Inserarea imaginilor în L^AT_EX

Este aproape inutil să explicăm ce importanță au cutiile trasate în jurul imaginilor PostScript. Aceste cutii sunt inserate de către T_EX printre cutiile proprii. Rezultatul este de o calitate deosebit de bună și este foarte ușor de obținut.

Voi descrie două cazuri: inserarea simplă a unei imagini PostScript și transformarea ei într-un corp flotant, la care ne putem referi cu ajutorul mecanismului trimiterilor din L^AT_EX.

Litera phi stilizată este inserată în textul de față cu ajutorul următoarelor rânduri din sursa L^AT_EX:

```
1 \begin{center}
2 \includegraphics[scale=1]{./imagini/phi.ps}
3 \end{center}
```

Mediul `center` n-are decât funcția pe care o sugerează numele său, aceea de a centra imaginea în raport cu textul. Dacă ați creat fișierul `phi.ps` îl puteți insera într-un document L^AT_EX și puteți experimenta schimbând valoarea opțiunii `scale`.

Cealaltă metodă este ilustrată cu ajutorul codului L^AT_EX folosit pentru a insera figura C.1:

```
1 \begin{figure}[ht]
2 \includegraphics[width=.98\textwidth]{./imagini/bitmap.ps}
3 \caption{0 imagine care folosește o hartă a pixelilor mărită}
4 \label{puncteBitmap}
5 \end{figure}
```

Mediul `figure` joacă același rol pentru imagini ca și mediul `table` pentru tabele. Observați succesiunea comenzilor din rândurile 2-4. Am folosit opțiunea care indică direct lățimea (ca procent din lățimea textului).

⁹Pentru a înțelege utilizarea fișierelor `jpg` vedeți §C.2.1.

Trimiterile la figuri se pot face folosind comanda `\ref{}` și eticheta definită în mediul `figure`.

Mediul `figure` creează un obiect care plutește prin text. Opțiunile sale `h` și `t` îi spun L^AT_EX să încerce fie o plasare chiar în punctul respectiv din text, fie în partea de sus a paginii.

În cazul eseurilor cu caracter academic aș pleda pentru acest mod sobru de inserare a figurilor în text. Mediul `wrapfig` permite îngroparea imaginilor în text, dar orice rescriere a textului dă peste cap toată munca de potrivire a textului în jurul figurii. Este o soluție pentru tehnoredactarea unui text care nu mai suferă schimbări. Se ciocnește de ideea de bază a elaborării textului academic: rescrierea repetată a textului.

Dacă vreți să aveți text alături de o imagine, fără prea multe complicații, puteți folosi mediul `minipage`. Cu ajutorul lui secționați pagina și plasați într-o parte text, în alta imaginea.¹⁰

C.2.1 Inserarea de imagini în fișiere pdf

Dacă PostScript este un limbaj orientat în special către tipărirea pe hârtie, ruda sa PDF este un limbaj orientat către producerea de cărți electronice. Un fișier `pdf` este aidoma unui text tipărit, dar este destinat lecturii pe un ecran de computer.

Limbajul PDF are propriile sale comenzi și, folosind compilatorul `pdflatex.exe`, puteți avea acces la ele. Există, de asemenea, o serie de pachete L^AT_EX orientate către producerea de fișiere `pdf`.

Un fișier `pdf` are două mari avantaje: este portabil (poate fi văzut atât sub Unix, cât și sub Windows); este compact și ușor de trimis sau de descărcat pe Internet. Cu alte cuvinte, este o superbă carte electronică.

Puteți converti direct un fișier `ps` într-unul `pdf` folosind Ghostscript. Recomandabil ar fi să folosiți opțiunea `LaTeX => PS => PDF` la compilare, disponibilă în versiunea 1 beta 6.20 a T_EXnicCenter.

Dezavantajul convertirii descrise mai sus este acela că n-aveți acces la compilatorul `pdflatex`. Limbajul PDF are specificul său, în raport cu PostScript, și este firesc să fie așa: o carte electronică se citește altfel decât una tipărită.

Există însă o problemă cu compilatorul `pdflatex`: nu cunoaște limbajul PostScript. Dacă aveți imagini PostScript, veți primi la compilare un mesaj de eroare: *unknown graphics extension*.

¹⁰ A se vedea aici efectul acestei soluții la pagina 246, rândul 5.

Există două soluții. Prima constă în convertirea separată a fișierelor PostScript în fișiere pdf și includerea lor ca pdf-uri. A doua constă în folosirea unui fișier jpg.

fișiere jpg Formatul jpg este de tipul *bitmap* (harta pixelilor).¹¹ Nu rezistă la scalare. Este, în schimb, un fișier comprimat. Fișierele *bitmap* pure tind să fie foarte mari. Fișierele *jpg* sunt de mici dimensiuni. Au și marele avantaj de a putea fi integrate în fișierul *dvi*, dacă faceți o manevră foarte simplă. 5

De exemplu, pentru a integra în *dvi* fișierul *phi.eps*, trebuie creat un fișier de tip text *phi.bb* (aceiași nume, dar extensia *bb*). Fișierul *bb* cuprinde un singur rând, o copie exactă a rândului din fișierul *ps* sau *eps* în care este declarată cutia, *BoundingBox*-ul. 10

Fișierul *jpg* poate fi integrat apoi și-n fișierul *pdf* final prin compilare cu *pdflatex*. Sfatul meu este să folosiți fișiere *jpg*. Oricât ar putea părea de curios, aceste fișiere sunt mai ușor de integrat în documentul final în limbaj PDF decât fișierele *pdf*. De asemenea, ele permit lucrul cu fișierul *dvi*. Este absolut nerecomandabil să lucrați pe parcurs cu fișierul *pdf*. 15

Realizarea fișierului *pdf* este o operațiune similară cu cea de tipărire pe un printer. Diferența constă doar în faptul că se folosește o „cerneală electronică” aplicată pe o foaie de „hârtie electronică”. Nu confundați sub nici o formă un fișier *pdf* cu unul de tip text sau cu un fișier *dvi*, care vă permite să regăsiți exact rândul din sursa *L^AT_EX* care a generat un anumit efect pe ecran. 20

C.3 Inserarea literelor ca inserare de imagini 25

T_EX este specializat în plasarea de cutii pe foaia de hârtie. Pentru ca să poată pune însă litere și simboluri în cutii trebuie să colaboreze cu *METAFONT*. Din această colaborare rezultă forța nebanuită a sistemului *T_EX*.

Ca și *T_EX*, *METAFONT* lucrează în fundal, ascuns în spatele perdeli de macro-uri *L^AT_EX*. 30

C.3.1 Desenarea unei litere

Limbajul *METAFONT* este orientat către *descrierea* la nivel *meta* a tipurilor de litere. Probabil că prea puține cititoare sau cititori se

¹¹Puteți crea fișiere *jpg* cu Gimp. A se vedea §C.2.1.

vor aventura prin hățişurile METAFONT. Prezentarea noastră este, în orice caz, pentru cei care vor să ştie câteva chestiuni de principiu.

Ca de obicei, ideea noastră este că modul cel mai bun de a învăţa (chiar şi principiile în materie de programare) este meşterirea
 5 unui program şi experimentarea. Creaţi cu Vim un fişier `phi.mf` cu următorul conţinut:

```

1 u#:=1/2pt#;
2 define_pixels(u);
3 beginchar(70,12u#,12u#,6u#);"Litera phi";
4   x1=x2=u; x3=x4=11u;
5   y1=y4=y5=7u; y2=y3=u;
6   x5=x6=6u;
7   bot y6=-5u;
8   pickup pencircle scaled 0.5pt;
9   draw z1..z2;
10  draw z2..z3;
11  draw z3..z4;
12  draw z4..z5;
13  draw z5..z6;
14  labels(range 1 thru 6);
15 endchar;
16 end

```

Ar fi destul de greu să ne dăm seama ce semnificaţie are programul `phi.mf` fără a genera o imagine. Deschideţi cu 2xExplorer o fereastră MS-DOS pentru dosarul în care se află `phi.mf` şi daţi două
 10 comenzi:

```

1 mf phi
2 gftodvi phi.2602gf

```

Prima comandă compilează fişierul. A doua comandă creează un fişier `dvi`.

Vizualizarea cu Yap a imaginii C.2 nu este o chestiune de rutină. Dacă aveţi un computer cu suficient de multă memorie, căutaţi în
 15 dosarul `misc` din `\texmf\fonts\source\public` fişierul `gray.mf` şi aduceţi singurul său rând la forma următoare:

```

1 input graylj % the 'standard' gray font is for the CX

```

Deschideţi Yap şi daţi un clic pe Options... din meniul View şi-n panoul Display alegeţi în caseta Mode un printer de tip LaserJet cu 600dpi.¹² Acum ar trebui să vedeţi bine imaginea în Yap.

¹²600dpi este o măsură a rezoluţiei printerului (*dots per inch*). Cum minus-culele pete (*dots*) de cerneală seamănă cu pixelii, măsura este bună şi pentru ecran.

METAFONT output 2004.02.05:1801 Page 1 Character 70 "Litera phi"

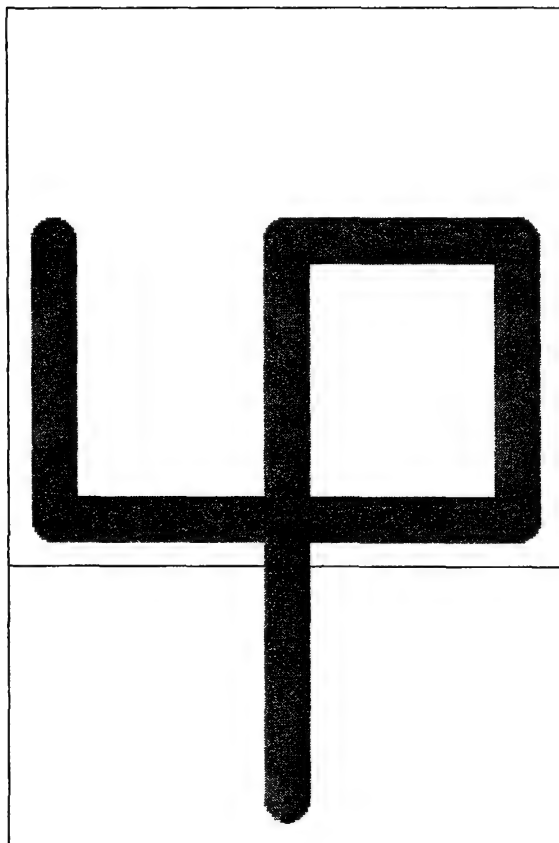


Figura C.2: Imagine de control pentru programe scrise în METAFONT

În orice caz, uitați-vă la Document Properties... din meniul File din Yap. Examinați tipurile de litere (*fonts*) încărcate și vedeți unde se găsesc pe computer. Citiți eventualele mesaje de eroare și soluțiile adoptate automat de către Yap.

5 Acum putem comenta, cu imaginea alături, conținutul fișierului `phi.mf`. Primul rând permite mărirea sau micșorarea desenului. Rândurile 4–7 conțin descrierea unor puncte din desenul literei. Numele punctelor sunt de forma `z_`, unde în locul spațiului subliniat este pusă o cifră. Rândurile 9–13 conțin descrierea liniilor. Rândul 10 8 indică grosimea liniilor.

Rândul 3 are o importanță aparte. În paranteza rotundă sunt patru valori. Prima este codul literei. Următoarele două valori indică poziția colțului din dreapta sus al cutiei literei (*bounding box*-ului). Ultima valoare indică poziționarea liniei de bază a literei. Culiile cu 15 litere sunt înșiruite de către \TeX în așa fel încât liniile de bază se continuă, pe orizontală, una pe alta.

Limbajul METAFONT este unul declarativ, descriptiv. Sistemului nu i se spune ce acțiuni să întreprindă, ci cum arată litera.

Structura programului este extrem de simplă. Pentru a vedea 20 efectele schimbării descrierii literei, modificați ceva din program și vedeți ce se întâmplă. S-ar putea să nu puteți compila deloc. Tastați `x` și încercați să scăpați. Nu efectuați însă decât o singură modificare o dată, pentru a localiza efectele.

Litera noastră este în mod intenționat construită din linii drepte. 25 Rotunjirile sunt posibile însă. Scrieți, în locul `draw z3..z4` și `draw z4..z5`, o declarație `draw z3..z4..z5` și veți descoperi ce deștept este METAFONT.

C.3.2 Inserarea literei în text

Pentru a ne folosi efectiv de o literă desenată trebuie să pregătim 30 fișierele de care se folosește \TeX . Trebuie să ne întoarcem în fereastra MS-DOS și să dăm comenzile:

```
1 mf \mode=ljfour; mode_setup; input phi.mf
2 gftopk phi.600gf phi.pk
```

Fișierele `phi.pk` și `phi.tfm` sunt cele care reprezintă descrierea literei în formatul de care se folosește \TeX . Acum pot scrie despre:

un φ cu colțuri.

Cum am procedat? Am pus fișierele de tip `mf pk` și `tfm` în dosarul proiectului cărții¹³ și am spus sistemului să recunoască un nou tip de literă și să-l folosească:

```
1 \newfont{\literaPhi}{phi}
2 \newcommand{\filos}{\{\literaPhi F\}}
3 un \filos{} cu colțuri.
```

Prezentarea de mai sus este una strict informativă. Litera construită n-are o valoare practică. În orice caz, valoarea ei este foarte limitată. Pot rescrie formula preluată din cartea lui Adrian Miroiu:¹⁴

$$\vdash_{LK} \varphi \text{ ddacă } \models_{LK} \varphi$$

Am folosit însă o cale ocolită. Pentru a insera, în modul matematic, simbolul φ l-am apelat în argumentul unei comenzi `\text{}`.5

Dac-ați meșterit însă variante ale fișierului `phi.mf` vă dați mult mai bine seama care este sensul mesajelor pe care le primiți la compilarea în `TEXnicCenter`. Acum este posibil să identificați mai ușor partea care vine de la `METAFONT` și cea care provine de la `TEX`.

Un lucru important pe care trebuie să-l știți este că `METAFONT`10 creează *bitmap*-uri. Micile desene care sunt literele sunt inserate în document ca imagini de tip hartă de pixeli. Hărțile acestea pot fi însă modificate în funcție de dimensiunea literelor. Puteți primi însă mesaje de genul „nu găsesc litera la dimensiunea cerută” sau să aveți dificultăți cu documentele `pdf` (litere neclare pe ecran) dacă folosiți15 doar nucleul `LATEX`.¹⁵

¹³Yap compilează automat sursele literelor; lipsa fișierului de tip `mf` poate duce la erori.

¹⁴Vezi aici pagina 268.

¹⁵Mai multe informații și exemple pe tema literelor și simbolurilor folosite în `LATEX` ne propunem să le plasăm, cu timpul, pe situl cărții.

Anexa D

Translatorii

\LaTeX este un limbaj pentru cei care au nostalgia cărților frumos tipărite. Este, de asemenea, limbajul ideal pentru elaborarea scrierilor cu caracter academic. În lumea de azi există însă multe alte limbaje pentru documentele electronice.

La începutul celui de al treilea mileniu, Internetul găzduiește deja o bibliotecă planetară. Limbajul său de bază este HTML. Editoarele de birou au propriile lor limbaje, cel mai adesea ascunse de privirea persoanelor care le utilizează, dar fără de care n-ar fi posibilă afișarea pe ecran și tipărirea textelor produse cu ajutorul lor.

Există deci o problemă a integrării documentelor \LaTeX în acest peisaj variat și, în funcție de necesități, a traducerii lor în alte limbaje.

Trebuie distins însă între două situații: traducerea părții care este în mod text și traducerea părții care este în mod matematic. În ambele cazuri, traducerea este dificilă. Deosebirea rezidă însă în faptul că, în primul caz, traducerea este – în linii mari – posibilă. În al doilea caz, traducerea este foarte greu de făcut.

Recomandarea noastră, pentru seminarii și examene, este aceea de a tipări documentele \LaTeX . La urma urmei, pentru asta sunt făcute.

Tipărirea poate fi pe hârtie sau electronică. În cazul tipăririi electronice, formatul pdf este cel la care trebuie ajuns, dacă este posibil. Utilizarea pachetului `hyperref`, creat de către Sebastian Rahtz, este, de asemenea, absolut recomandabilă în cazul tipăririi electronice.¹

¹A se vedea explicațiile lui Sebastian Rahtz, în *Hypertext marks in \LaTeX* , fișierul `hyperref.dvi` din `\texmf\doc\latex\hyperref`, precum și restul documentației din dosarul respectiv. Nu încercați să înțelegeți de la bun început

Date fiind scopurile cărții de față, prezentarea în detalii a traducerii din \LaTeX în alte limbaje n-are sens. Obiectivul nostru principal îl reprezintă producerea unor texte tipărite. Această anexă oferă doar câteva informații cu privire la posibilitățile și tehnicile de traducere, fără nici un fel de indicații cu caracter practic. Pentru instalarea, utilizarea și modificarea programelor la care ne vom referi trebuie să consultați documentația lor. 5

Nu uitați niciodată să efectuați fiecare traducere într-un dosar separat. Pentru a obține rezultate mai bune sursele \LaTeX trebuie pregătite în mod special în vederea traducerii, conform indicațiilor din documentația relevantă. 10

D.1 Drumul către HTML

Documentele \LaTeX , tipărite electronic în fișiere **ps** sau **pdf** sunt ușor de vehiculat pe Internet. Așa cum am arătat mai sus, formatul **pdf** este preferabil. 15

Limbajul HTML este însă limbajul de bază al Internet-ului. Dacă vrem să prezentăm eseurile noastre pe Internet, trebuie să știm câte ceva despre HTML.

Puțină lume știe că HTML este o aplicație, cu posibilități limitate, a SGML. Acronimul SGML vine de la *Standard Generalized Markup Language*.² Rațiunea existenței SGML poate fi mai lesne înțeleasă dacă vom reaminti modul de lucru tradițional într-o editură. 20

Redactorii și tehnoredactorii unei edituri scriau pe dactilograma textului care urma să fie tipărit o serie întreagă de indicații. Unele dintre ele priveau tipul de literă ce urma să fie folosit în tipografie. Altele priveau poziția unei porțiuni din text în raport cu restul textului și așa mai departe. În limba engleză, toate aceste operații sunt desemnate cu termenul *mark up*. 25

Ideea de bază a SGML este simplă. Porțiunile de text în care urmează să se aplice o anumită indicație sunt puse într-o paranteză construită în felul următor: 30

`<tag>porțiunea de text aflată între etichete</tag>`

Fiecare **tag** are un nume. Se observă mai sus cum începutul porțiunii etichetate este marcat de numele **tag**-ului pus între paranteze

subtilitățile tehnice ale pachetului; simpla lui includere în document și câteva minime adaptări ale documentului vor produce efecte spectaculoase.

²Bradley[5, p.4] arată că SGML a devenit un standard în 1986.

ascuțite, iar finalul ei este semnalat prin bara oblică pusă înaintea tag-ului.

Numele tag-urilor, precum și indicațiile de prelucrare a porțiunilor de text plasate sub cupola unui tag sunt date separat în *Document Type Definition*, pe scurt DTD.

DTD

Pentru ca perechea formată dintr-un text în care sunt incluse indicații în SGML și un DTD să conducă și la efecte concrete pe ecranul unui computer este nevoie și de un analizor (*parser*). Analizorul detectează construcțiile SGML din text și le prelucrează conform instrucțiunilor din DTD. Analizorul poate să fie, de pildă, un procesor de cuvinte sau un program de vizualizare a textelor structurate conform SGML.

Spre deosebire de \TeX , SGML nu este un limbaj de programare. Este un limbaj destinat identificării *obiectelor* dintr-un text. Printre aceste obiecte se poate afla însă și o porțiune de cod \TeX ³ sau o secvență scrisă în alt limbaj de programare.

Mecanismul trimiterii electronice care pot fi definite în SGML este cunoscut sub numele de „hipertext”. Hipertextul nu este altceva decât o dezvoltare a ideii tradiționale de trimitere. În mod tradițional, dacă o notă conținea o referire la o carte, ca să vezi efectiv pasajul din carte la care se făcea trimitere te deplasai la bibliotecă, cereai cartea și citeai fragmentul cu pricina. Prin contrast, trimiterea electronică nu indică doar locul unde se poate citi ceva, ci chiar îți aduce pe ecran textul la care se face trimitere.

hiper-
textul

HTML este o aplicație a limbajului SGML. Deschideți cu *Vim* un fișier de tip `html` și veți remarca imediat tag-urile, scoase în evidență cu ajutorul culorilor. Studiați, de asemenea, primul rând al fișierului și ar trebui să vedeți ceva de genul:

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
```

Observați precizarea privitoare la DTD. Chiar dacă ea lipsește, vizualizatorul de HTML se descurcă. Analizările de HTML nu sunt nici pe departe așa de pretențioase cu textul pe care-l au în față ca \TeX .

Capacitățile HTML sunt destul de limitate, dar în ultima vreme există un efort considerabil de a dezvolta XML, în încercarea de a redescoperi întreaga putere a SGML.

Din descrierea sumară de mai sus se vede că \LaTeX și HTML au și o serie de puncte comune. Sunt, în orice caz, perfect compatibile.

³Bradley[5, p.72] explică felul în care poate fi identificată porțiunea scrisă în \TeX .

Soluția mea pentru un sit academic ar fi o serie de pagini HTML interconectate cu ajutorul mecanismului hipertextului. În ele se găsesc trimiteri electronice la fișiere pdf sau ps.⁴ O parte dintre documentele L^AT_EX sunt traduse în HTML; unele bibliografii sunt oferite și împreună cu fișierul bib aferent.

De ce să nu scriem direct în HTML? În cazul eseurilor academice, L^AT_EX creează automat bibliografia, notele de subsol, indicii. HTML n-are resursele unui limbaj de programare și tot aparatul auxiliar academic ar trebui creat manual. Evident, este posibil să recurgem la alt limbaj de programare care să completeze cumva capacitățile HTML (să țină, de pildă, socoteala numerelor notelor de subsol); L^AT_EX este însă soluția superioară la acest capitol.

D.1.1 De la L^AT_EX la HTML cu ajutorul H^EV^EA

H^EV^EA este un program creat de către Luc Maranget. Philip A. Viton a realizat o versiune a H^EV^EA sub Windows.⁵

Instalarea H^EV^EA sub Windows se face fără nici o dificultate, dacă urmați instrucțiunile lui Philip A. Viton.

Integrarea în T_EXnicCenter nu este nici ea dificilă. Imitați profilul folosit la crearea unui fișier ps, dar cu hevea.exe ca postprocesor și cu Internet Explorer ca vizualizator.

Rezultatele obținute cu H^EV^EA sunt bune și, la o sursă L^AT_EX simplă, aproape că n-aveți ce schimba în fișierul html. La textele în limba română aveți grijă ca următorul tag din fișierul html să aibă **charset** attributele corecte:

```
1 <META http-equiv="Content-Type" content="text/html;
2                                     charset=windows-1250">
```

Valoarea lui **charset** poate să fie și ISO-8859-2; windows-1250 pare mai naturală aici, dacă documentul L^AT_EX folosește cp1250; fără aceste valori ale lui **charset** nu se văd literele românești. Nu este recomandabil să puneți utf-8 ca valoare a **charset**. Pentru limbile europene n-aveți nevoie de întreaga putere a unicode. Pagina dumneavoastră se va încărca lent din pricina dimensiunilor fișierului cu simboluri unicode.

Dacă doriți să folosiți entități HTML, puteți încerca transformarea fișierului de tip html cu ajutorul unui script Vim ca acesta:

⁴Pentru strategia integrării documentelor pdf în rețeaua de informații de pe web v. McKinley[27].

⁵Pentru a porni în căutarea H^EV^EA mergeți la pagina de Internet a lui Philip A. Viton la <<http://www.arch.ohio-state.edu/crp/faculty/pviton/>>.

```

1 :%sno/ă/&atilde;/ge | update
2 :%sno/Ă/&Atilde;/ge | update
3 :%sno/â/&acirc;/ge | update
4 :%sno/Â/&Acirc;/ge | update
5 :%sno/î/&icirc;/ge | update
6 :%sno/Î/&Icirc;/ge | update
7 :%sno/ș/&scedil;/ge | update
8 :%sno/Ș/&Scedil;/ge | update
9 :%sno/ț/&tcedil;/ge | update
10 :%sno/Ț/&Tcedil;/ge | update

```

Puteți pune și coduri numerice, după moda XML, prin substituiți ca acelea de mai jos:

```

1 :%sno/&atilde;/&#259;/ge | update
2 :%sno/&Atilde;/&#258;/ge | update
3 :%sno/&acirc;/&#226;/ge | update
4 :%sno/&Acirc;/&#194;/ge | update
5 :%sno/&icirc;/&#238;/ge | update
6 :%sno/&Icirc;/&#206;/ge | update
7 :%sno/&scedil;/&#351;/ge | update
8 :%sno/&Scedil;/&#350;/ge | update
9 :%sno/&tcedil;/&#355;/ge | update
10 :%sno/&Tcedil;/&#354;/ge | update

```

D.1.1.1 Pregătirea documentului pentru Internet folosind HACHA

- 5 Documentele HTML, oricât de lungi ar fi, sunt derulate sub forma unei pagini unice pe ecranul computerului. Acest lucru are două dezavantaje: când conexiunea Internet este slabă, încărcarea paginii se face anevoios; de asemenea, este dificil să urmărești o pagină care se derulează foarte mult.
- 10 Ideal ar fi ca pagina să ocupe aproximativ un ecran. În orice caz, plasarea secțiunilor unui document în fișiere html diferite este dezirabilă.

Programul care vă ajută să faceți secționările descrise mai sus este HACHA. Acest program trebuie folosit după ce ați prelucrat documentul cu ajutorul HEVEA.

D.1.1.2 HEVEA și imaginile din documentele HTML

Programul HEVEA a fost creat de către Luc Maranget pentru a fi folosit sub Unix. Sub Unix el prelucrează automat imaginile la care face apel sursa L^AT_EX.

Sub Windows, fără uneltele Unix, trebuie să pregătiți manual imaginile pentru Internet. Veți găsi totuși, în fișierul `html`, create de către HEVEA, locurile în care se pot integra imaginile.

Dacă ați instalat unelte Unix și aveți la dispoziție *shell*-ul Unix, atunci trebuie să rescrieți scriptul `imagen` în așa fel încât să poată rula sub Windows. În aceste condiții, puteți include imagini în documente HTML și sub Windows.⁶

D.1.2 De la \LaTeX la HTML cu ajutorul \TeX4ht

Distribuția Mik \TeX include un translator de `html`. Dacă nu i-ați sesizat deja prezența, mergeți în dosarul `\texmf\tex4ht`. Acolo veți găsi o parte din elementele pachetului creat de către Eitan Gurari pentru a traduce din \LaTeX în HTML.

Dacă citiți descrierea pachetului \TeX4ht , veți vedea că este *configurabil*. Luați în serios diferența dintre *configurat* și *configurabil*. Ca peste tot în lumea \TeX , și aici se aplică principiul „Dumnezeu dă omului, dar în traistă nu-i bagă”, așa că trebuie să configurați \TeX4ht . Philip A. Viton are o pagină de sprijin pentru persoanele care instalează \TeX4ht în cadrul Mik \TeX .⁷

Dac-ar fi să dau câteva indicații bazate pe experiența proprie, aş spune, în primul rând, c-am mutat fișierele din dosarul `base`, aflat pe calea `\texmf\tex4ht` în dosarul `home`. Trebuie modificat cu grijă fișierul `tex4ht.env`. Atenție trebuie acordată și fișierului `tex4ht.fls`; dacă nu există, trebuie creat unul de tip text cu acest nume în dosarul unde sunt plasate `texmf` și `localtexmf`.

Pentru a vedea dac-ați reușit să configurați \TeX4ht faceți un test cu un fișier `tex`. Eu am folosit sursa manualului lui Stéphane Bressan și Eitan Gurari pentru pachetul $\text{\TeX}Project$.⁸ Am instalat elementele din pachetul respectiv. Am pus sursa manualului într-un dosar în care nu era alt fișier. Am adăugat în sursă comanda `\usepackage{tex4ht}` și am dat într-o fereastră MS-DOS comanda `ht latex manual`. Fără a exagera, se poate spune că rezultatul a fost de o calitate deosebit de bună.

⁶Pentru descrierea modificărilor operate de către noi în scriptul `imagen` vizitați situl cărții. În momentul de față, versiunea noastră convertește, sub Windows, imaginile PostScript și le integrează în documente HTML.

⁷Adresa paginii cu explicațiile lui Philip A. Viton este <http://facweb.arch.ohio-state.edu/pviton/support/tex4ht.html>.

⁸Pachetul respectiv poate fi descărcat de pe pagina de web a lui Eitan Gurari <http://www.cis.ohio-state.edu/~gurari/index5.html>.

TeX4ht poate produce și fișiere XML în formatul definit de către DTD-uri ale OpenOffice.

D.1.3 De la L^AT_EX la HTML cu ajutorul L^AT_EX2HTML

L^AT_EX2HTML a fost conceput inițial de către Nikos Drakos. Opera sa a fost continuată de către Ross More și alții. Ideea lor de bază este aceea de a realiza un translator care traduce în întregime documentul L^AT_EX: unele părți sunt traduse direct în limbajul HTML, iar ceea ce nu se poate traduce este transformat în imagine și inserat ca imagine în documentul HTML.

Recursul masiv la imagini are și unele dezavantaje. Dacă descărcați un text convertit din L^AT_EX în HTML cu L^AT_EX2HTML trebuie să recurgeți la `wget`.⁹ Altfel, aproape sigur veți rămâne cu fără unele imagini sau hipertextul va fi stricat. De asemenea, din pricina imaginilor căutarea unora dintre termeni este imposibilă sau trebuie făcută cumva indirect (folosind elemente din fișierul HTML care, în mod normal, nu se văd).

Instalarea L^AT_EX2HTML sub Linux nu cred că mi-a luat mai mult de cinci minute. Iar programul a funcționat fără cusur. În schimb, sub Windows, ar trebui să fiți fericiți sau fericiți dacă duceți la bun sfârșit instalarea.

În principiu, pentru instalare, aveți nevoie de Perl și de Net-PBM.¹⁰ Perl este un limbaj de programare. L^AT_EX2HTML este un script realizat în Perl. Din pricina aceasta nu intru aici în detalii.¹¹

În rezumat, programul este dificil de instalat sub Windows, poate să vă lase fără resurse de memorie și nu este tocmai ușor de folosit. Este totuși cel mai vestit dintre programele de traducere din L^AT_EX în HTML și simplul fapt de a-l putea folosi sub Windows va oferi oricui o mare satisfacție. Este recomandabilă folosirea lui în cazul transformărilor mai dificile. Este singurul dintre translatorii care s-a descurcat cu anexele cărții de față chiar și fără să fi operat vreo schimbare în sursa L^AT_EX.

⁹A se vedea aici §A.4.2.1.

¹⁰A se vedea Luis Seidel Gómez de Quero, *Installing L^AT_EX2HTML with MikTeX* <<http://www.mayer.dial.pipex.com/12h.htm>>, 4 februarie 2000, document accesat pe data de 20/08/2003.

¹¹A se vedea situl cărții pentru detalii.

D.2 Drumul către RTF

Formatul RTF este proprietatea firmei Microsoft. A fost creat pentru a facilita transferul de texte formate între diverse aplicații, sub diverse sisteme de operare. Între altele, el deschide drumul către popularul format Word al firmei Microsoft.

Dacă o editură sau o revistă vă solicită textul în format Word, iar sursa dumneavoastră este în \LaTeX , aveți la dispoziție două soluții. Puteți folosi HTML ca pe o platformă de pe care să treceți la formatul Word. Puteți folosi în același scop RTF. Pentru aceasta vă trebuie însă un translator din \LaTeX în RTF.

Cea mai bună alegere este programul \LaTeX2RTF disponibil pe marele sit al programării cu sursă deschisă, SourceForge.net.¹² Începând cu 20 ianuarie 2004 există și un program de instalare sub Windows. Dacă nu instalați în dosarul care vi se propune în mod automat, trebuie să efectuați manual unele corecturi în fișierele de tip bat.

Pentru a beneficia de toată puterea \LaTeX2RTF trebuie să aveți instalat Ghostscript și ImageMagick¹³.

Sintaxa RTF seamănă, sub anumite aspecte, cu sintaxa \TeX .¹⁴ Iată un fragment de cod RTF:

```
% Aceasta este {\b o carte} foarte grea. \par
```

Porțiunea prinsă între acolade reprezintă un grup. Ca și o instrucțiune \TeX , secvența `\b` îi spune programului care interpretează fișierul `rtf` că grupul trebuie scris cu aldine. Alternativ, am putea folosi comenzile `\b` și `\b0` pentru a marca începutul, respectiv sfârșitul porțiunii de text care este scrisă îngroșat.

Puteți experimenta pe un fișier `rtf` creat cu ajutorul translatorului folosind editorul Vim. Fișierele de tip `rtf` sunt fișiere text.

RTF are un mod de a trata căciula de pe ă și sedilele lui ș și ț care n-arată prea estetic în programul de vizualizare.¹⁵ Eu ocolesc

¹² Adresa exactă este <http://latex2rtf.sourceforge.net>.

¹³ ImageMagick este un program de convertire de imagini. Pentru mai multe informații vizitați situl <http://www.ImageMagick.org>. Identificați cu atenție în programul instalat locul în care se află programul `convert.exe`.

¹⁴ Pentru detalii consultați pagina <http://latex2rtf.sourceforge.net/rtfspec.html>.

¹⁵ Pentru a vedea un fișier `rtf` puteți folosi programul de vizualizare a documentelor Word pus la dispoziție gratuit de către Microsoft. Acest program poate citi și fișiere `rtf`. Puteți construi propriul lector de `rtf` folosind codul inclus de către Microsoft în specificația limbajului RTF. Ca și formatul RTF, programul

traducerea automată în stil rtf substituind o parte dintre literele românești în felul următor:

```
1 :%s/\\u{a}/a227/ge | update
2 :%s/\\u{A}/A195/ge | update
3 :%s/\\~{a}/â/ge | update
4 :%s/\\~{A}/Ă/ge | update
5 :%s/\\~{i}/î/ge | update
6 :%s/\\~{I}/Î/ge | update
7 :%s/\\c{s}/s186/ge | update
8 :%s/\\c{S}/S170/ge | update
9 :%s/\\c{t}/t254/ge | update
10 :%s/\\c{T}/T222/ge | update
```

În documentul final, aceste substituiri sunt eliminate. Atenție însă la acest subterfugiu dacă aveți într-un tabel pe A195 și nu vreți să devină Ă.

Fișierul rtf n-ar trebui predat ca atare redacției care l-a solicitat. El necesită aproape întotdeauna unele prelucrări suplimentare. El este un fișier destinat transferului între aplicații. Redacției ar trebui să-i fie predat un fișier în formatul propriu aplicației pe care o folosește.

D.3 Înapoi către L^AT_EX

Ar trebui spuse câteva cuvinte și despre posibilitatea de a traduce în L^AT_EX formatările din alte limbaje. Problema s-ar putea să se pună practic cel mai frecvent în cazul documentelor Word.

Formatul Word al firmei Microsoft este foarte popular. Imensa majoritate a persoanelor care utilizează computerele nu au motive să investească timp și bani în învățarea programării (chiar într-o măsură limitată). Editorul Word al firmei Microsoft a reprezentat o reușită deosebită în efortul de a ascunde aspectele legate, direct sau indirect, de programare. Au dispărut comenzile cu punct din editoarele mai vechi. Utilizatorii sunt scutiți de orice contact cu marcajele introduse în text. Aceasta este rațiunea succesului produsului respectiv pe piață.

Trecerea de la Word la L^AT_EX este departe de a fi chiar atât de simplă. Există o serie de produse comerciale care încearcă să facă – fără prea mult succes – acest lucru. Eu mă voi referi aici

de vizualizare și codul pentru lector sunt proprietatea firmei Microsoft și pot fi folosite numai în condițiile respectării drepturilor firmei. Citiți desigur licențele aferente.

la un program cu sursă deschisă, MSWordView.¹⁶ MSWordView a fost conceput inițial ca un program de traducere, sub Unix, a documentelor Word în HTML. În stadiul actual este capabil să facă o traducere, inclusiv sub Windows, în \LaTeX .

Partea bună a MSWordView atunci când traduce în \LaTeX este capacitatea sa de a produce o sursă \LaTeX curată. Redarea literelor specifice limbii române este, de asemenea, impecabilă. 5

Partea proastă a lucrurilor începe atunci când este vorba despre notele de subsol. MSWordView marchează locul lor în text și mută tot textul notelor la sfârșit. Este perfect posibil să înlocuiești marcajul $\text{\$}\{ \}\text{\$}$ introdus de translator cu comanda $\text{\backslash footnote}\{ \}$ și să muți apoi textul notelor, dar munca este anevoioasă. 10

În rezumat, MSWordView este util dacă aveți un text masiv, cu puține note de subsol, pe care vreți să-l tehnoredactați în \LaTeX . 15

Cocluzia este limpede. \LaTeX este un limbaj pentru elaborarea unor surse organizate logic. Este un punct de plecare, nu o țintă de atins plecând de la alte moduri de a structura texte. 15

Faptul că sursa \LaTeX este practic extrem de greu de reconstituit are o consecință de care se pot folosi persoanele care se tem că lucrările lor ar putea fi plagiate cu ușurință dacă sunt plasate pe Internet. Ele pot pot afișa doar fișierele care pot fi citite comod. Sursele nu sunt făcute publice. Ar fi foarte greu pentru plagiatori să pretindă că sunt adevărații autori. Reconstituirea absolut exactă a surselor originare este practic imposibilă. Dacă ținem cont de faptul că sursele originare conțin în mod normal și comentarii, reconstituirea este absolut imposibilă. 20 25

¹⁶Versiunea cea mai comod de instalat sub Windows este cea disponibilă în cadrul proiectului GnuWin32. A se vedea <http://gnuwin32.sourceforge.net>.

Bibliografie

- [1] Robert R. Arnold, Harold C. Hill și Aylmer V. Nichols. *Inițiere în prelucrarea datelor*. București: Editura Tehnică, 1969. Citată la pp. 29 și 30.
- [2] Charles Bazerman. *The Informed Writer: Using Sources in the Disciplines*. Boston: Houghton Mifflin Company, 1989. Citată la p. 17.
- [3] Vasile Biriș. *Tehnoredactare: cărți, reviste, ziare*. București: Editura AXY, 2003. Citată la p. 25.
- [4] Paul A. Blaga și Horia F. Pop. *L^AT_EX 2_ε*. București: Editura Tehnică, 1999. Citată la pp. 184, 269, 273 și 275.
- [5] Neil Bradley. *The concise <SGML> companion*. Harlow, Anglia: Addison-Wesley, 1996. Citată la pp. 290 și 291.
- [6] Barry Brown. *How to Write an Essay in Bioethics* [indicații pentru lucrările studentești]. <<http://www.chass.utoronto.ca/philosophy/phlwrite/brown1.html>>, 1996, data accesării: 21/06/2003. Citată la p. 7.
- [7] University of Chicago Press. *A Manual of Style*, ediția a 12-a. Chicago: The University of Chicago Press, 1969. Citată la pp. 68, 69, 71, 72 și 80.
- [8] Melvin Dewey. *Decimal Classification and Relative Index*. New York: Forrest Press, 1959. Revizuită de Comitetul pentru clasificarea zecimală Dewey de la Biblioteca Congresului SUA. Citată la p. 9.
- [9] Sergiu Drincu. *Semnele ortografice și de punctuație în limba română*. București: Editura Științifică și Enciclopedică, 1983. Citată la pp. 96 și 99.

- [10] E. I. Dubovik (Дубовик), A. A. Veize (Бейзе) și S. A. Golovko (Головко). *Реферирование и аннотирование экономических текстов на английском языке*. Minsk: Višeišiaia škola, 1977. Citată la pp. 22 și 57.
- [11] Daniel N. Fischel. *A Practical Guide to Writing and Publishing Professional Books: Business, Technical, Scientific, Scholarly*. New York: Van Nostrand Reinhold Company, 1984. Citată la p. 95.
- [12] Antony Flew. Issues in teaching contemporary ethics. *Teaching Philosophy*, 1(1):55–60, vara 1975. Citată la p. 65.
- [13] Antony Flew. *Thinking about Thinking*. Glasgow: Fontana/Collins, 1975. Citată la pp. 24, 62, 73 și 96.
- [14] I. Funeriu. *Principii și norme de tehnoredactare computerizată*. Timișoara: AMARCORD, 1998. Citată la pp. 16, 21, 25, 79 și 205.
- [15] Al. Graur. *Mic tratat de ortografie*. București: Editura Științifică, 1974. Citată la pp. 96, 97, 98, 99, 100 și 110.
- [16] Diana Hacker. *Research and Documentation Online*. <<http://www.dianahacker.com/resdoc/index.html>>, data accesării: 11/08/2003. Citată la pp. 117 și 118.
- [17] Johannes Erich Heyde. *Technik des wissenschaftlichen Arbeitens*. Berlin: Kiepert, 1966. Citată la pp. 12 și 13.
- [18] Peter T. Horban. *Writing a Philosophy Paper*. <<http://www.sfu.ca/philosophy/writing.htm>>, 1993, data accesării: 11/08/2003. Citată la pp. 7 și 66.
- [19] Victor Iancu și Săluc Horvat. *Dicționar de ortograme*. București: Editura Științifică, 1994. Citată la pp. 21, 73, 109 și 110.
- [20] R. R. Jordan. *Academic Writing Course*. Londra: Collins, 1980. Citată la p. 117.
- [21] Leslie Lamport. *L^AT_EX: a document preparation system*. Reading, Massachusetts: Addison Wesley Longman, Inc., 1994. Citată la pp. 184, 211, 212, 227 și 273.
- [22] James D. Lester, Sr. și James D. Lester, Jr. *The Research Paper Handbook*. Glenview, Illinois: Scott, Foresman and Company, 1992. Citată la pp. 72, 88, 95 și 117.

- [23] Evan Mawdsley și Thomas Munck. *Számítógép a történettudományban: kalauz kezdőknek* [orig. *Computing for Historians: an introductory guide*]. Budapesta: Osiris, 1996. Citată la p. 85.
- [24] James M. McCrimmon. *Writing with a Purpose*. Boston: Houghton Mifflin, 1967. Citată la pp. 57, 58 și 63.
- [25] Ashley McDowell. *How to Write a Philosophy Paper*. <<http://www.u.arizona.edu/~ashleym/writing.htm>>, 1998–2002, data accesării: 11/08/2003. Citată la p. 7.
- [26] Henry McGilton și Mary Campione. *PostScript by Example*. Reading, Massachusetts: Addison-Wesley, 1992. Citată la p. 280.
- [27] Tony McKinley. *From Paper to Web*. San Jose, California: Adobe Press, 1997. Citată la p. 292.
- [28] A. I. Mihailov, A. I. Ceornîi și R. S. Ghilearevski. *Informatica documentară*. București: Editura Științifică, 1970. Citată la pp. 5, 37 și 49.
- [29] Bram Moolenaar. *Vim User Manual*. <<http://vimdoc.sf.net>>, 2002, data accesării: 30/12/2003. Citată la pp. 155, 157, 158, 163, 165 și 166.
- [30] Costică Moroșanu. *Elemente de bază ale sistemului T_EX și plain T_EX*. Iași: Editura Universității „Alexandru Ioan Cuza”, 2001. Citată la pp. 21, 183 și 184.
- [31] Ioan Oprea. *Terminologia filozofică românească modernă*. București: Editura Științifică, 1996. Citată la pp. 105 și 106.
- [32] Steve Oualline. *Vi IMproved—Vim*. <<http://www.truth.sk/vim/vimbook-OPL.pdf>>, 2001, data accesării: 30/12/2003. Citată la p. 155.
- [33] Roger C. Parker. *Looking Good in Print*. Scottsdale, Arizona: Coriolis, 1998. Citată la p. 60.
- [34] William C. Paxson. *The Business Writing Handbook*. New York: Bantam Books, 1984. Citată la pp. 39 și 95.
- [35] Douglas W. Portmore. *Tips on Writing a Philosophy Paper* [ghid electronic pe situl autorului la California State University, Northridge]. <<http://www.csun.edu/~dp56722/tips.htm>>, 2001, data accesării: 22/02/2004. Citată la pp. 7 și 116.

- [36] James Pryor. *Guidelines on Writing a Philosophy Paper* [pe situl lui James Pryor la Princeton]. <<http://www.princeton.edu/~jimpryor/general/writing.html>>, 2003, data accesării: 21/06/2003. Citată la pp. 7, 46 și 62.
- [37] Psychology Writing Center. *Style Points for Scientific Writing* [pe situl University of Washington]. <<http://depts.washington.edu/pswyc/handouts/style.html>>, data accesării: 21/06/2003. Citată la p. 58.
- [38] Artur Pusztai și Gheorghe Ardelean. *L^AT_EX: ghid de utilizare*. București: Editura Tehnică, 1994. Citată la p. 184.
- [39] Raymond Seroul. *Le petit Livre de T_EX*. Paris: InterEditions, 1989. Citată la pp. 28, 202, 203, 228 și 229.
- [40] A. W. Sparkes. *Talking Philosophy: a Wordbook*. Londra: Routledge, 1991. Citată la p. 59.
- [41] William Strunk, Jr. *The Elements of Style*. New York: 1918. <<http://www.columbia.edu/acis/bartleby/strunk/>>. Citată la pp. 22, 58 și 59.
- [42] Peter Suber. *Essay Assignment* [ghid pentru Earlham College, Richmond, Indiana]. <<http://www.earlham.edu/~peters/courses/essay.htm>>, 1999–2002, data accesării: 22/02/2004. Citată la pp. 7, 19 și 33.
- [43] Sun Microsystems. *Read Me First! A Style Guide for the Computer Industry*. Mountain View, California: SunSoft Press, 1996. Citată la pp. 79 și 86.
- [44] Barbu Theodorescu. *Istoria bibliografiei române*. București: Editura Enciclopedică Română, 1972. Citată la p. 39.
- [45] S. Tóth. *Litera de tipar*. București: Editura Științifică, 1966. Citată la p. 184.
- [46] John Vickers. *Cum să vă indexați cartea* [carte electronică pe situl cimec București] (traducere de Doina Anca Cornaciu). <<http://www.cimec.ro/Carte/indexare/index.htm>>, 1996, data accesării: 10/03/2001. Citată la pp. 77 și 78.
- [47] Martin Young. *Philosophical Writing Manual* [pe situl dedicat „undergraduate philosophical writing” la University of California at Irvine]. <<http://eee.uci.edu/programs/philoswr/long.htm>>, data accesării: 22/11/1999. Citată la p. 7.

Indice

- §, *vezi* secțiune, 208
- alineat, *vezi* paragraf
- apud*, 88
- argumente, 61, 62
- pro și contra, 65, 66
 - pseudoargumentul autorității, 64
 - reconstituire, 24
- AUTOEXEC.BAT, 139, 140
- betacod, 256–259
- BibDB, 237
- bibliografie, 32, 34, 36, 38, 39, 49
- alcătuită manual, 67–70
 - cu $\text{BIB}\text{T}\text{E}\text{X}$, 238–240
 - stil, 15, 238, 239
 - stil simplu, 239
- bitmap, *vezi* imagini(hărți de pixeli)
- Bozinis, Nikos, 128
- Braams, Johannes L., 243
- bun de tipar, 16
- căutare, *vezi* și expresii regulate
- în fișierul tradițional, 36, 37
 - pe Internet, 16, 174–180
- cale. 125
- CIP, 11
- clasificare. 8
- Biblioteca Congresului, 9
 - Dewey, 9
 - zecimală, 10
- computerul și munca intelectuală, 29, 30, 190, 191
- conținut vs. formă, 27, 51, 232
- concordanță, 173, 174
- corectura
- computerizată, 180–182
 - manuală, 108–112
- ctags, 221–223
- cuprins
- alcătuire manuală, 49
 - cu $\text{L}\text{A}\text{T}\text{E}\text{X}$, 200, 272
- cuvinte-cheie, 20, 23, 78
- Denisov, Alex, 264
- denumirile produselor, 85
- descriptor, 77, 78, 276
- distincții filosofice, 63
- Doron, Eyal, 237
- Drakos, Nikos, 295
- empatia, 84
- eseu, 3–5
- academic vs. propagandă, 54, 96
 - alte universități, 117
 - claritate vs. stil, 56
 - elaborare, 52
 - evaluare, 114–117
 - limbajul, 95, 96
 - principiul sobrietății, 83
 - rolul problemelor, 53, 97
 - sustinere, 113, 114
- exemple, 65
- explicații, 64
- expresii grecești, 104

- ul style="list-style-type: none;">
- expresii latinești, 101–103
- expresii regulate, 166–172
- extrasele, 88
 - în note, 93, 94
 - în textul de bază, 93
 - lungimea lor, 89
 - punctuația, 94
 - trunchiate, 93
- fișe, 12, 127
 - bibliografice, 13–17, 233, 237
 - de lectură, 17–20
 - de lucru, 20
 - formatele, 12, 13
- fișiere
 - analogia cu fișele tradiționale, 127
 - bb, 284
 - bib, 235
 - bmp, 278
 - bst, 238
 - căutarea, 131
 - comprimate, 132
 - crearea, 129
 - csv, 233, 234
 - dvi, 194, 198, 199
 - extensiile, 128
 - filtre, 130
 - html, 291
 - jpg, 284
 - log, 204
 - management cu 2xExplorer, 128
 - mf, 285
 - numărarea cuvintelor, 140
 - operații, 130
 - pdf, 134, 283
 - proprietăți, 130
 - ps, 133, 134, 280–282
 - redenumire, 137
 - rtf, 296
 - sty, 228
 - stergere, 137
 - tex, 197
 - tiff, 138
 - xcf, 282
- filozofie vs. filosofie, 110
- formule, 218, 219, 267–274
- Garfield, Eugene, 37, 38
- gcc, 141
- Ghostscript, 133
- Gimp, 279, 281
- GNU, 142
- GnuWin32, 141
- Gonzato, Guido, 231
- grep, 171
- Gurari, Eitan, 294
- HACHA, 293
- HEVEA, 292
- hieroglifice, 262
- HOME, 139
- Ibycus, 259, 260
- idei-cheie, *vezi și* propozițiile-cheie
 - în această carte, 45, 46
- imagini
 - hărți de pixeli, 278, 279
 - vectoriale, 280, 281
- indicele citărilor, 38
- indici
 - generare manuală, 49, 77, 78
 - generarea lor cu MakeIndex, 276
- indici vs. cuprins, 77
- ISBN, 11
- ISSN, 11
- Jacq, Christian, 262
- Joy, William, 144
- Knuth, Donald, 25, 79, 183–188, 203, 242, 254

- Lamport, Leslie, 25, 184
 LaTable, 264
 L^AT_EX, 184–186
 #, 224
 alineat, 205
 antet, 196
 bad box(es), 215
 \code{}, 256
 {\bfseries }, 212
 blocurile, 211
 \cite{}, 240
 comentarii, 206
 compilare, 197
 contoare, 226
 corpul programului, 198
 cp1250, 246–249
 declarațiile, 212
 diacritice, 243
 documentație, 231
 {\em }, 212
 \fontfamily{}, 253
 \fontsize{}, 253
 \footnote{}, 214
 ghilimele, 210
 \hspace{}, 215
 hyperref, 289
 ifthen, 227
 \index{}, 276
 {\itshape }, 212
 \label{}, 207
 \ldots{}, 210
 letrine, 228
 \marginpar{}, 214
 \mbox{}, 211
 {\mdseries }, 212
 mediile, 216
 modul matematic, 218
 modurile, 203
 \newcommand{} {}, 223
 \newenvironment. 225
 pachete, 226
 pachete proprii. 228
 \pageref{}. 220
 \par{}, 205
 \ref{}, 220
 \relax, 229
 revizia, 212
 {\rmfamily }, 212
 româna, 242
 {\scshape }, 212
 secțiuni, 206
 \selectfont{}, 253
 semne rezervate, 195, 209, 210
 {\sffamily }, 212
 {\slshape }, 212
 structura paginii, 230
 \textbf{}, 209
 \textit{}, 209
 \textrm{}, 209
 \textsc{}, 209
 \textsf{}, 209
 \textsl{}, 209
 \today{}, 211
 {\ttfamily }, 212
 \underline{}, 209
 unicod, 260
 {\upshape }, 212
 \vspace{}, 216
 L^AT_EX2HTML, 295
 L^AT_EX2RTF, 296
 lectura
 lentă, 23
 rapidă, 23
 Levy, Silvio, 259
 litere
 chirilice, 254, 255
 grecești, 255–260
 limbile europene, 251, 252
 românești, 24, 149, 241–245, 270
 Łukasiewicz, Jan, 54
 MacKay, Pierre, 259
 manuscrisul
 ideile-cheie, 45

- modulele, 45–48
- Maranget, Luc, 292
- matchit, 202
- menționare, 59, 60
 - limbaje de programare, 60
- meniu contextual, 124
- METAFONT, 263, 284–287
- MikTeX, 188, 189
- Miller, George, 47
- More, Ross, 295
- MS-DOS, 135–137
- MSWordView, 298
- neutralitate axiologică, 55
- note
 - de subsol, 70–75, 214
 - marginale, 214
- octet, 153, 154
- Oetiker, Tobias, 231, 232
- OpenOffice, 180, 295
 - folosirea bibliografiei, 144
- Ostrum, Piet van, 232
- Pakin, Scott, 270
- parafrizarea, 88
- paragraf, 18, 21, 22, 27, 57, 75, 206
 - ascendent, 22
 - cu împarantizare, 22, 58
 - cu cheie implicită, 22, 58
 - descendent, 22
 - organizarea logică, 59
 - pivotant, 22, 58
- partițiile, 125
- Patashnik, Oren, 238, 239
- Pehong Chen, 275
- pixel, 277
- plagiatul, 90–93, 298
- probleme, 53, 62
 - ocolirea vicioasă, 63, 64
 - punctuale, 54
- proiect
 - L^AT_EX, 199–201
 - de eseu, 31–35, 114
 - versiuni, 48
- propoziție, 57
 - activ vs. pasiv, 58
 - concizia, 58
- propoziție-cheie, 20, 22
- publicul cititor, 39–41
- punctul, 97
- rând
 - logic, 59, 148
 - sfârșitul de rând, 127, 155
 - vizual, 148
- Rahtz, Sebastian, 289
- rescrierea textului, 25, 46
- rezumarea, 20–24
- secțiune, 18, 21, 22, 206
- semnul exclamării, 98
- SGML, 290
- sit, 177
- splitfile, 126
- stil bibliografic
 - APA, 118
 - Chicago, 69, 118, 239
 - edituri românești, 72
 - MLA, 117
 - simplu, 70, 143, 239, 240
- subdescriptor, 78, 276
- substituție, 170
- surse
 - primare, 5, 6
 - secundare, 7
- surse deschise, 142, 143
- tabele, 219, 263–267, 271, 273, 274
- temă vs. comentariu, 21
- termenii filosofici, 105–108
- T_EX4ht, 294, 295
- T_EXnicCenter, 171, 172, 177, 191–193
- T_EXnicCenter
 - și Vim, 193

- ul style="list-style-type: none; padding-left: 0;">
- titlurile, 84, 85
- trimiteri, 50
 - în comentarii, 222
 - în pagini web, 177, 291
 - de la un indice la altul, 276
 - la altă pagină, 76, 220
 - la altă secțiune, 76, 220
 - la bibliografie, 240
 - la figuri, 283
 - la tabele, 265
 - la un rând, 76
- ucenici vrăjitori, 28
- unicod, 260, 292
- Unix, 125, 139, *vezi și* Gnu-Win32
 - unelte, 140
- Unruh, Dominique, 261
- utilizare, 59, 60
 - vs. menționare, 60
- utilizare vs. programare, 25–29, 213
- versiuni, *vezi* rescrierea textului
- Vim
 - căutare, 172, 173
 - comenzi în linie, 147
 - configurare, 149
 - corectura sintactică, 151
 - cursorul, 160, 161
 - dialog, 217
 - instalare, 145
 - meniuri, 156–159
 - modificarea textului, 162–165
 - modul insert, 145, 146
 - modul normal, 145, 146
 - modul vizual, 146
 - scripturi, 150
 - semne de carte, 152
 - tastaturi, 250
 - virgula, 98–101
 - Viton, Philip, 292
 - Volovich, Vladimir, 242
 - wget, 178, 179
 - Wilson, Peter, 262
 - Windows
 - ascuns, 137–139
 - comenzi, 135
 - ghidul, 122–125
 - windows-1250, 292
 - WindowsXP
 - variabile de mediu, 140
 - Winograd, Terry, 21
 - Word, 297
 - Yap, 194, 199, 216, 285, 287, 288
 - și Vim, 194

Tiparul s-a executat sub cda nr. 1200/2004
la Tipografia Editurii Universității din București

Eseul filosofic

Cartea este un îndrumar pentru lucrările de „laborator” la filosofie. Este vorba în special despre lucrările de seminar, dar ea poate fi folosită pentru elaborarea unor lucrări mai complexe.

Sunt tratate diferitele faze ale elaborării unui eseu, de la proiectul de cercetare și documentarea inițială până la prelucrarea informației, testarea ideilor proprii și redactarea finală.

Accentul este pus, în carte, pe documentare și redactarea finală, care sunt mai ușoare de așezat sub semnul unor reguli precise.

O mare atenție este acordată acelor faze în care se poate folosi în mod intens calculatorul. Este indicat modul de a obține, în texte de factură academică, pe printerul de acasă sau din laboratorul facultății, rezultate apropiate de cele pe care le au editori care folosesc personal specializat, echipamente și programe scumpe. Există însă un preț în spatele a orice. Prețul care trebuie plătit constă în efortul de a învăța câteva elemente de bază ale programării orientate către prelucrarea de texte.

Mihail Radu Solcan
Universitatea din București
Facultatea de Filosofie
<http://www.fil.unibuc.ro/~solcan/eft/>